# Web-Based Slide Decks

Fabian Platzer

706.415 Seminar Interactive and Visual Information Systems
Graz University of Technology

09 Dec 2024

**Abstract**

This survey explores various tools for creating web-based slide decks, focusing on four categories: text-based, JavaScript-based, hosted, and responsive. Each tool is summarised and its key features are highlighted. The tools are then compared to other tools in the same category, offering insights into their suitability for different presentation needs. The survey aims to provide a comprehensive overview to help guide users and developers select the most appropriate slide deck tool for their requirements.

# Contents

# List of Figures

# List of Tables

# List of Listings

# Chapter 1

# Introduction

The evolution of web-based slide decks has significantly transformed the way presentations are created and shared. The pioneering tools in this space were Slidy [Raggett 2005] and Slidy2 [Raggett 2006], which introduced the idea of HTML-based slide decks as an alternative to traditional tools such as Microsoft PowerPoint [Gaskins et al. 2024] and Apple Keynote [Apple 2003].

This survey gives an overview of web-based slide deck tools for presentations. It builds upon Patrick Hipp's comprehensive 2019 survey [Hipp 2019] and groups web-based slide decks into four categories: text-based, JavaScript-based, hosted, and responsive. Each category is described in a dedicated chapter.

To better compare the tools, a listing and a figure of the same example slide was created with every tool at the same window size (1600×900). The example slide has three bullet points on the left and a vector graphic (in SVG format) on the right. If an SVG graphic cannot be displayed by a particular tool, a PNG image is used instead. In most cases, custom CSS styling is used to achieve a two-column slide layout. Where the CSS styling is too extensive, it has been omitted from the example listings for brevity.

All the slide deck tools are compared across the following 17 criteria:

- *Collaborative*: Whether the tool allows multiple users to work on a presentation simultaneously.
- *SVG Inclusion*: Whether the slide deck tool supports inclusion of SVG graphics.
- *Globally Change Bullet Spacing*: Ability to modify the vertical spacing between bullet points throughout the whole presentation.
- *Globally Change Bullet Indentation*: Ability to adjust bullet point indentation across all slides.
- *Export as PDF*: Whether the tool supports exporting the presentation as a PDF file.
- *Export as HTML*: Whether the tool supports exporting the presentation in an HTML format.
- *Export as PowerPoint*: Ability to export the presentation in Microsoft PowerPoint format.
- *Slide Numbering*: Support for automated slide numbering in the presentation.
- *Live Code Integration*: Support for embedding of live code within the presentation.
- *Open-Source*: Whether the tool is open-source, allowing for customisation and modification.
- *Licence*: The licensing model, indicating whether the tool is proprietary or open for public use.
- *First Release*: The first known release of a tool.
- *Last Update*: Most recent update of the tool, reflecting its ongoing development.
- *Popularity (Usage)*: The level of popularity and usage of the tool. Measured by GitHub stars where

appropriate, otherwise estimated.

- *Swipe Navigation*: Whether the tool supports swiping gestures to navigate forwards and backwards between slides.
- *Margin Navigation*: Whether the tool supports tapping or clicking the slide margins to navigate forwards and backwards between slides.
- *Figure Zooming and Panning*: The ability to zoom in and out and pan around in embedded figures (images and graphics).
- *Presenter Mode*: Whether the slide deck tool offers a presenter mode with a second window for speaker notes.

A comparison table and a short discussion is included at the end of each of the following four chapters.

## 1.1 Slidy [2005-2006]

Slidy is an HTML-based slide presentation tool developed by Dave Raggett [Raggett 2005], a member of the W3C team and a pioneer in web standards. Slidy was designed to create slide shows that could be viewed directly in web browsers. It generates a slide-show from a single XHTML file, which includes a CSS file for styling and a JavaScript file for the functionality. Each slide is enclosed in a `<div class ="slide">` element. An example can be seen in Figure 1.1; the corresponding source code is shown in Listing 1.1. The key features of Slidy include:

+ *Based on HTML and CSS*: Easy customisation and styling using standard web technologies.
+ *JavaScript-Enhanced*: Interactive features like navigation controls, slide transitions, and multimedia embedding.
+ *Keyboard Shortcuts*: Easy slide navigation during presentations.
+ *Incremental Display*: Bullet items revealed incrementally.
+ *SVG Inclusion*: Include vector graphics as SVG.

Slidy's use of HTML and CSS for structure and styling made it highly appealing to web developers.

## 1.2 Slidy2[2006-2013]

Building on the foundation laid by Slidy, Slidy2 introduced several enhancements to improve usability and functionality [Raggett 2006]. The key enhancements include:

+ *Improved User Interface*: Improved navigation controls and user-friendly interface elements.
+ *Advanced Features*: Support for more complex animations, better multimedia integration, and responsive design for desktop screen sizes.
+ *Customisation Options*: Greater flexibility in customising the appearance and behaviour of slide decks through extended CSS and JavaScript support.

Although Slidy2's enhancements addressed some of the limitations of the original Slidy, the design was still optimised for desktop usage, which brings its own limitations and reduced functionality when used on mobile devices with smaller screens.

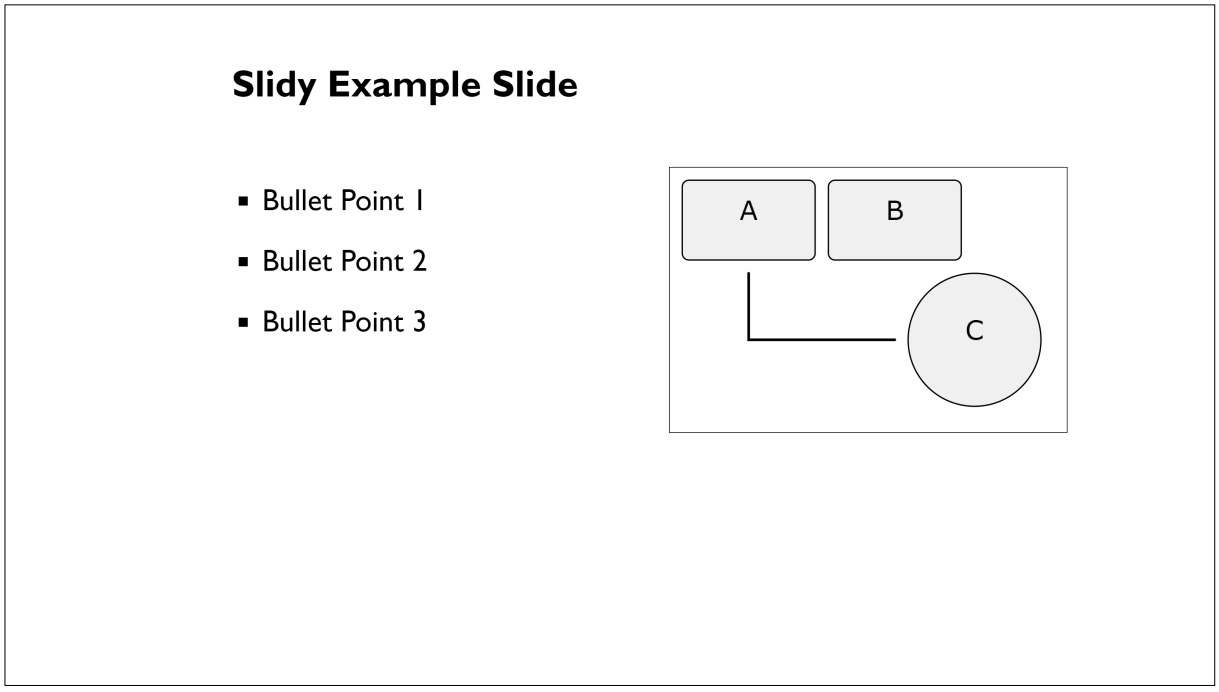


**Figure 1.1:** Slidy: Example slide. The corresponding source code is shown in Listing 1.1.

```
<div class="slide">
  <h1>Slidy Example Slide</h1>
  <div class="container">
    <div class="left-column">
      <ul>
        <li>Bullet Point 1</li>
        <li>Bullet Point 2</li>
        <li>Bullet Point 3</li>
      </ul>
    </div>
    <div class="right-column">
      <img src="graphics/diagram.svg" alt="Example Image">
    </div>
  </div>
</div>

<style>
  .container {
    display: flex;
    justify-content: space-between;
    align-items: flex-start;
  }
  .left-column {
    width: 45%;
  }
  .right-column {
    width: 45%;
  }
  img {
    max-width: 75%;
    height: auto;
    margin-top: 3rem;
  }
</style>
```

**Listing 1.1:** Slidy: Example slide source code. The resulting slide is shown in Figure 1.1.

# Chapter 2

# Text-Based Slide Decks

Text-based slide decks provide an efficient and user-friendly alternative to traditional presentation tools. At the core of many text-based slide deck tools is Markdown [Cone 2024], a lightweight markup language first proposed by Gruber [2004]. Markdown allows users to write content in plain text, using an easy-to-read, easy-to-write format, which is then converted to structurally valid XHTML (or HTML). Using Markdown requires little technical expertise. Users can focus on content creation, without needing deep knowledge of HTML, CSS, or JavaScript.

This chapter describes four popular tools, created in a Node-based environment [Node 2024], for building text-based slide decks using Markdown: Remark, Marp, Fusuma, and Slidev. Section 2.5, at the end of the chapter, compares the tools.

Other systems, like Slide Show (S9) [Bauer 2011], Slidedown [Nakajima and Croak 2012], and Slidifier [Ludvigsen and Halvorsen 2011], were once widely used, but are now considered outdated. This survey does not include these systems, but refers the reader back to the survey by Hipp [2019].

## 2.1 Remark [2011-2023]

Remark is a simple, browser-based slide deck tool designed for users familiar with HTML and CSS, integrating both HTML and Markdown [Bang 2023].

To get started with Remark, the user needs to create an HTML file for the slide deck, or use the provided boilerplate template, open it in a browser, and edit the Markdown or CSS as needed. To enhance the experience, keyboard shortcuts, such as "C" to clone a display, "P" to switch to presenter mode, and "H" to access the help menu, are available.

In a Remark HTML file, style definitions are placed in the HTML header. The body includes a < `textarea id="source"`> element, where the Markdown text or HTML for the slides is placed. Calling the `create` function triggers the creation of a new slide deck, as can be seen in Figure 2.1, the corresponding source code is shown in Listing 2.1.

The key characteristics of Remark include:

+ *Markdown Formatting*: Markdown formatting with smart extensions for enhanced slide creation.
+ *Presenter Mode*: Features speaker notes and a cloned slideshow view for presentations.
+ *Syntax Highlighting*: Supports a variety of programming languages with syntax highlighting.
+ *Custom Templates*: Markdown templates for customised slide designs.
+ *Swipe Navigation*: Supports touch gestures support on smartphones and tablets.
- *Collaboration*: Does not support built-in collaboration features.

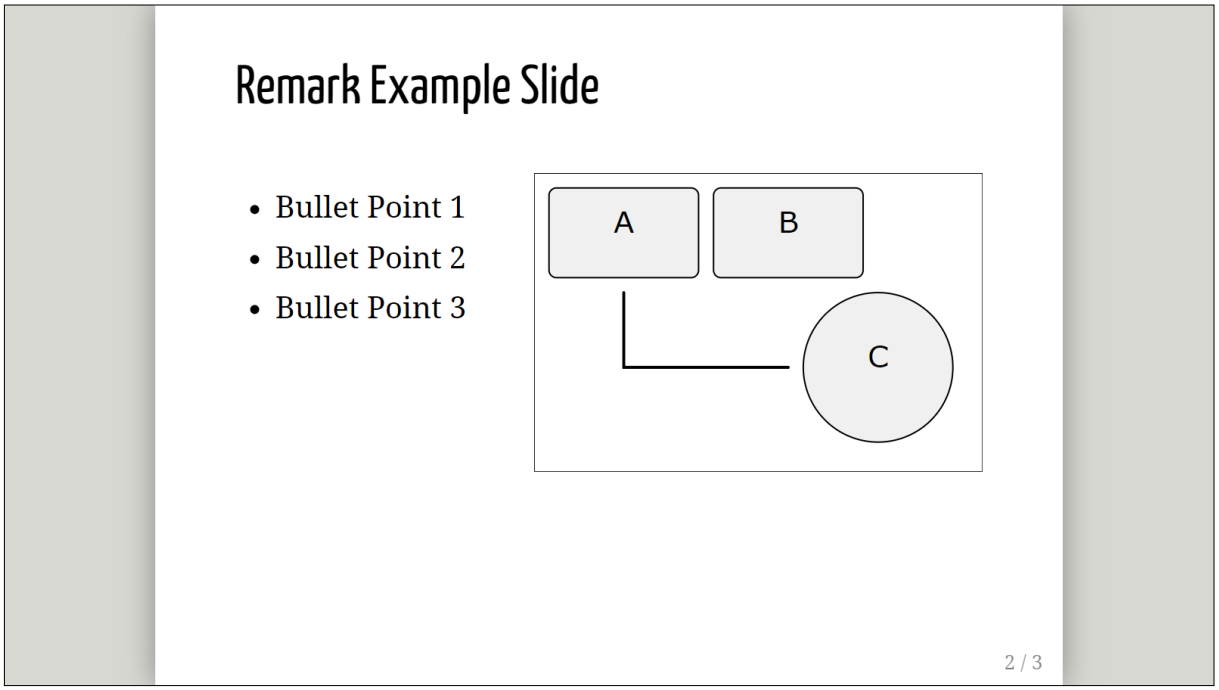


**Figure 2.1:** Remark: Example slide. The corresponding source code is shown in Listing 2.1.

- *Export Options*: Not possible to export presentations as PowerPoint files.
- *Live Code Integration*: Does not offer live code integration.

## 2.2 Marp [2018-]

Marp combines the simplicity of Markdown with the flexibility of HTML, offering features such as slide-specific styles, presenter notes, and a presenter view [Hattori 2024]. Slides are separated using three dashes "`---`", and presenter notes are added with XML comments enclosed in "`<!--`" and "`-->`".

Marp's input format is an extension of a standard Markdown variant called CommonMark [MacFarlane et al. 2024]. A Marp input file starts with YAML front matter enclosed by three dashes "`---`", which allows for global slide settings such as themes and transitions. The body of the document contains the Markdown text for the slides. HTML elements can also be mixed in. A simple example can be seen in Figure 2.2; the corresponding source code is shown in Listing 2.2. Marp can then be run on the command line with the command:

```
marp deck.md
```

or as a plugin for Visual Studio Code.

The key characteristics of Marp include:

+ *Slide Preview*: Instant preview of slides.
+ *Custom Themes*: Simple and easy modifiable slide design.
+ *IDE Integration*: Extension for Visual Studio Code [Marp 2023].
+ *Advanced Features*: Syntax highlighting and maths typesetting with MathJax [AMS 2022] and KaTeX [Khan 2024].
+ *Multiple Export Formats*: Export to various formats including PDF, HTML, and PowerPoint.

```
<body>
<textarea id="source">
---
# Remark Example Slide

.left-column[
* Bullet Point 1
* Bullet Point 2
* Bullet Point 3
]

.right-column[
  ![Image Description](./diagram.svg)
]

---
</textarea>
<script src="https://remarkjs.com/downloads/remark-latest.min.js">
</script>
<script>
    var slideshow = remark.create();
</script>
</body>

<style>
.left-column{
  width: 40%;
  float: left;
  line-height: 3rem;
  font-size: 2rem;
}

.right-column{
  width: 60%;
  float: right;

}
#img {
  width: 12.5rem;
  height: 6rem;
}
</style>
```

**Listing 2.1:** Remark: Example slide source created with a boilerplate.html template from GitHub [Bang 2023]. The source Markdown to create the slide deck must be inside a `<textarea>` element. Calling the `create()` function creates a new slideshow. The resulting slide is shown in Figure 2.1.

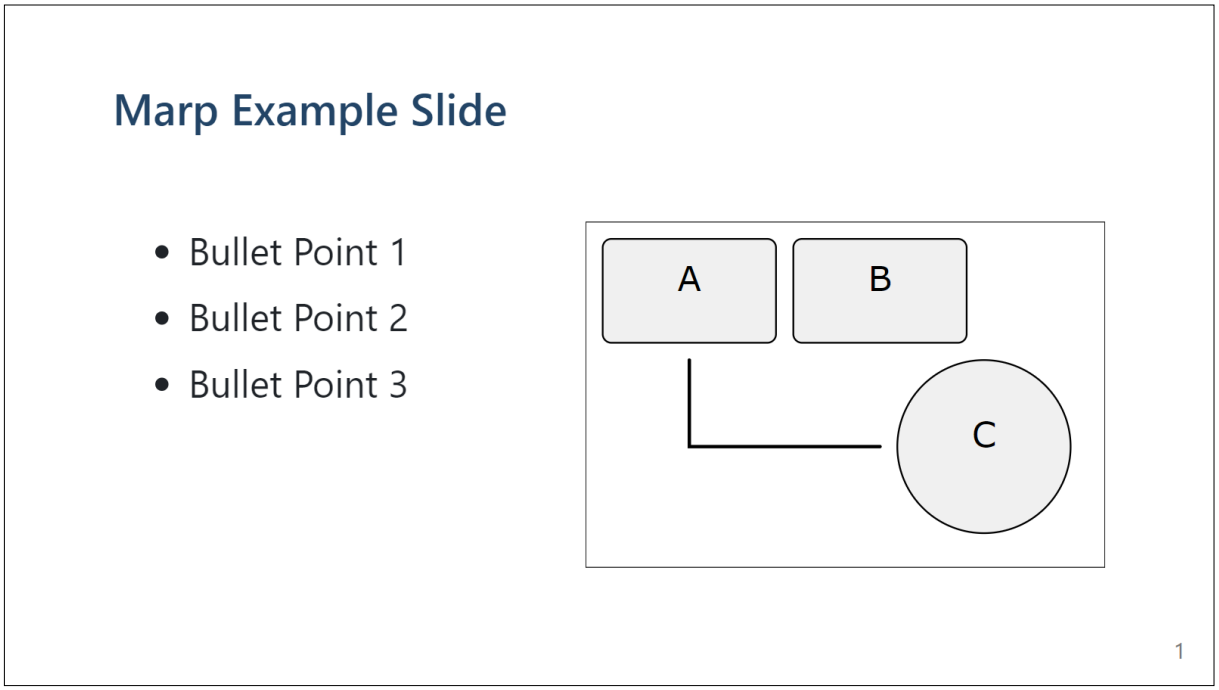


**Figure 2.2:** Marp: Example slide. The corresponding source code is shown in Listing 2.2.

+ *SVG Inclusion*: Supports the import of SVG vector graphics.
- *Collaboration*: Marp does not support collaboration features.
- *Swipe Navigation*: No swipe gestures for slide navigation.
- *Margin Navigation*: No margin tapping or clicking for slide navigation.
- *Live Code Integration*: Does not offer live code integration.

## 2.3 Fusuma [2018-2021]

Fusuma is a text-based online slide deck tool for Node [Node 2024], that allows users to create web-based presentations by editing plain text or Markdown files [Hiroto 2021]. Fusuma supports MDX [MDX 2018], which is a format that lets the user use JSX [Meta 2022] in Markdown documents. Fusuma can be installed over the command line with the command:

```
npm i fusuma -D
```

The command:

```
npx fusuma init
```

creates the project structure and the server can then be run with the command:

```
npx fusuma start
```

As in many other Markdown slide deck tools, slides in Fusuma are divided by three "`---`". Speaker notes can be added with "`<!-- note`", these can then be seen in the presenter mode. An example slide can be seen in Figure 2.3, with the corresponding source code in Listing 2.3.

The key characteristics of Fusuma include:

+ *Markdown and MDX Support*: Flexible content creation with Markdown and MDX, integrating JSX components.

```
---
marp: true
theme: default
paginate: true
backgroundColor: #fff
---

# Marp Example Slide
<!-- _class: split -->

<div class=ldiv>
- Bullet Point 1
- Bullet Point 2
- Bullet Point 3
</div>

<div class=rdiv>
![Example Image](./diagrams/diagram.svg)
</div>

<style>
  section.split {
  overflow: visible;
  display: grid;
  grid-template-columns: 31rem 34rem;
  grid-template-rows: 9rem auto;
  grid-template-areas:
      "slideheading slideheading"
      "leftpanel rightpanel";
  }

  section.split h3,
  section.split .ldiv,
  section.split .rdiv { border: 0px}
  section.split h3 {
      grid-area: slideheading;
      font-size: 3rem;
  }
  section.split .ldiv { grid-area: leftpanel; font-size: 2.5rem;}
  section.split .rdiv { grid-area: rightpanel; }
</style>
```

**Listing 2.2:** Marp: Example slide source code. Custom CSS styling is used to achieve the two-column layout for the example slide. The resulting slide is shown in Figure 2.2.

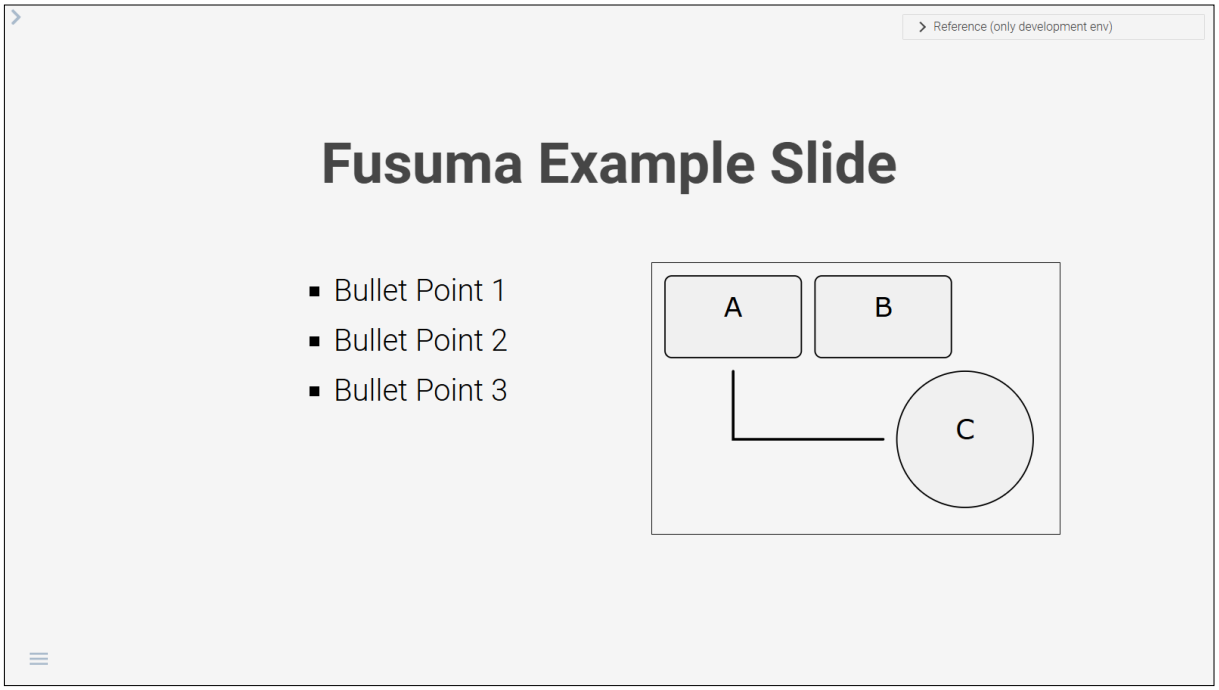


**Figure 2.3:** Fusuma: Example slide. The corresponding source code is shown in Listing 2.3.

+ *Custom Themes*: Wide range of themes for presentation customisation.
+ *Advanced Features*: Includes custom themes, syntax highlighting, and math typesetting with MathJax [AMS 2022].
+ *Sidebar with Agenda*: Sidebar features an agenda and useful presentation tools.
+ *Swipe Navigation*: Includes swipe and margin-tap gestures for navigation.
- *Collaboration*: Lacks built-in collaboration features.
- *Export Options*: Does not support HTML export.
- *Slide Numbering*: No automated slide numbering.
- *Live Code Integration*: Does not offer live code integration.

## 2.4 Slidev [2021-]

Slidev is a modern, web-based presentation tool using Markdown [Fu 2024]. It stands out for its extensive features, flexibility, and interactivity with the support of Vue.js [You 2024] components, that can be directly used in the slides. As an open-source tool with an active community, it also benefits from continuous support and development. Slidev can be started right in the browser with StackBlitz: sli.dev/new [StackBlitz 2024], or locally with npm, pnpm or yarn. Slidev is installed locally with the npm command:

```
npm init slidev@latest
```

The slide deck can then be built with the commands:

```
npm run dev
npm run build
```

```
---
Fusuma Example Slide
---
<div style={{ display: "flex", justifyContent: "space-between"}}>

<div style={{ width: "50%" }}>
  - Bullet Point 1
  - Bullet Point 2
  - Bullet Point 3
</div>

<div style={{ width: "40%" }}>
  <img src="./diagram.svg" alt="Example Diagram"/>
</div>

</div>
```

**Listing 2.3:** Fusuma: Example slide source code. Written in Markdown with CSS styling to achieve two columns for the example slide. The resulting slide is shown in Figure 2.3.

The slide deck can also be exported to PDF with the command:

```
npm run export
```

and also to PowerPoint and PNG with the command:

```
npm run export --format [format]
```

Slides are seperated by "`---`" padded with a new line, as can be seen in Figure 2.4 and the corresponding Listing 2.4.

The key characteristics of Slidev include:

+ *Developer Friendly*: Excellent support and documentation for developers, backed by a large community.
+ *Recording Support*: Built-in recording and camera view for separate screen and camera recording.
+ *Hackable*: Web-based nature allows all web app functionalities in presentations.
+ *Custom Themes*: Wide range of themes for presentation customisation.
+ *Interactive*: Integrates Vue.js components for interactive presentations.
+ *Live Code Integration*: Offers live code integration.
+ *Multiple Export Formats*: Simple slide export to formats such as PDF, PowerPoint, PNGs with one command.
- *Export as HTML*: Not possible to export presentations as HTML files.
- *Limited Offline Functionality*: Advanced features often require an online setup or third-party integrations.
- *Figure Zooming and Panning*: Lacks built-in functionality for zooming and panning figures.

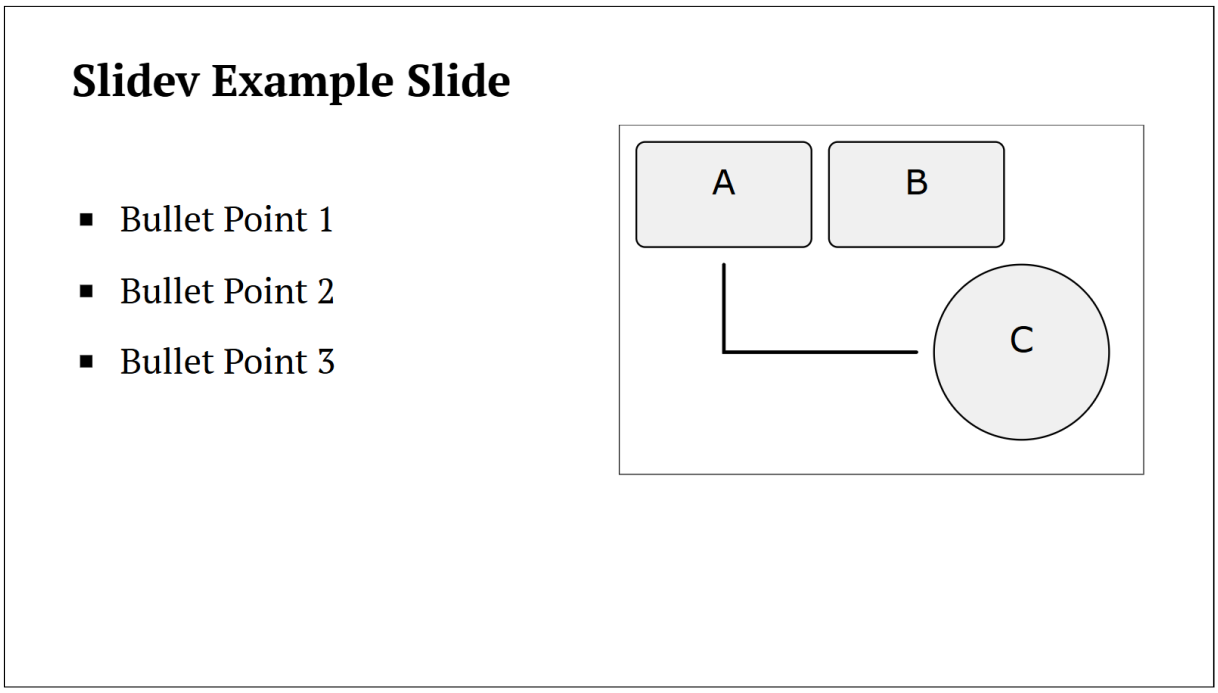


**Figure 2.4:** Slidev: Example slide. The corresponding source code is shown in Listing 2.4.

```
1  ---
2  layout: two-cols
3  ---
4  # Slidev Example Slide
5  - Bullet Point 1
6  - Bullet Point 2
7  - Bullet Point 3
8
9  ::right::
10 #
11 <img src="diagrams/diagram.svg" alt="Example SVG">
```

**Listing 2.4:** Slidev: Example slide source code written in Markdown. The layout `two-cols` separates the page content in two columns. The resulting slide is shown in Figure 2.4.

| | Remark | Marp | Fusuma | Slidev |
|---|---|---|---|---|
| Collaborative: | ✗ | ✗ | ✗ | ✗ |
| SVG Inclusion: | ✓ | ✓ | ✓ | ✓ |
| Globally Change Bullet Spacing: | ✗ | ✓ | ✗ | ✓ |
| Globally Change Bullet Indentation: | ✗ | ✓ | ✗ | ✓ |
| Export as PDF: | ✓ | ✓ | ✓ | ✓ |
| Export as HTML: | ✓ | ✓ | ✗ | ✗ |
| Export as PowerPoint: | ✗ | ✓ | ✓ | ✓ |
| Slide Numbering: | ✓ | ✓ | ✗ | ✓ |
| Live Code Integration: | ✗ | ✗ | ✗ | ✓ |
| Open-Source: | ✓ | ✓ | ✓ | ✓ |
| Licence: | MIT | MIT | MIT | MIT |
| First Release: | 2011-10-15 | 2018-03-25 | 2018-04-27 | 2021-04-12 |
| Last Update: | 2024-06-24 | 2024-04-21 | 2021-12-01 | 2024-07-22 |
| Popularity (GitHub stars): | 7.4k | 7.4k | 5.4k | 32.1k |
| Swipe Navigation: | ✓ | ✗ | ✓ | ✓ |
| Margin Navigation: | ✓ | ✗ | ✓ | ✓ |
| Figure Zooming and Panning: | ✗ | ✗ | ✗ | ✗ |
| Presenter Mode: | ✓ | ✓ | ✓ | ✓ |

**Table 2.1:** Comparison of text-based slide deck tools.

## 2.5 Comparison of Text-Based Slide Decks

A comparison of text-based slide deck tools is shown in Table 2.1. Marp, Fusuma, Remark, and Slidev, all support the inclusion of SVG graphics and exporting the slide deck to PDF. While only Marp and Remark offer HTML export, Marp, Fusuma, and Slidev offer the export of the slide deck as a PowerPoint presentation. However, these tools lack stand-alone collaborative capabilities and only offer limited customisation options. Global changes to bullet indentation and spacing can only be done with Marp and Slidev. This can be achieved with styling in a CSS file or the file itself. Only Slidev provides live coding integration, making it particularly useful for dynamic code presentations.

All tools are open-source with MIT licenses and have varying levels of popularity, with Slidev being the most popular. Only Marp fails to support swipe and margin tap navigation. All of the tools provide a presenter mode with a second window for speaker notes, enhancing the convenience for presenters. Compared to JavaScript-based slide decks, these tools are easy to set up with Markdown syntax and provide many key features for presentations like easy export to PDF and some user-friendly but limited customisation of slides.

# Chapter 3

# JavaScript-Based Slide Decks

JavaScript-based slide deck tools have transformed the way presentations are created by using the power of HTML, CSS, and JavaScript to create interactive, and customisable slides with animations, and interactions that are not typically available in static or traditional slide tools. These presentation are rendered in the browser, making it accessible from various devices with internet access. Most JavaScript-based slide deck tools, are available as packages on NPM, a package manager, that allows developers to manage and install JavaScript packages for Node.js projects [NPM 2024a].

This chapter covers six popular tools: Shower, Reveal.js, Deck.js, impress.js, Bespoke.js, and Inspire.js. All of these tools are node based, except for Deck.js. It gives an overview of each tool and then compares them based on specific criteria, showing insights into their features, functionality, and overall effectiveness for creating JavaScript-based presentations. Section 3.7, at the end of the chapter, compares the tools.

## 3.1 Shower [2010-]

Shower is a minimalistic HTML presentation tool that focuses on providing essential features without overwhelming users with complex options and features [Makeev 2024]. It imitates the appearance of traditional slide decks like PowerPoint. Slides are created with an `<section class="slide">` element, as shown in Figure 3.1 and the corresponding Listing 3.1. Presentations can be hosted on Netlify, a web hosting platform that simplifies building, deploying, and managing static websites web applications directly from a Git repository [Netlify 2024], as well as self-hosted with NPM. Self-hosting includes a gulp configuration file with predefined tasks such as “prepare”, which builds the slide deck, and “publish”, which uploads it to Netlify. They key characteristics of Shower include:

+ *Minimalist Design*: Focuses on minimalistic design and content rather than complex visual effects.
+ *Responsive Layout*: Adapts to different screen sizes and devices.
+ *Interactive Navigation*: Presentations can be navigated using keyboard shortcuts, as well as touch gestures and mouse clicks.
+ *PDF Export*: Presentations can be exported to PDF.
- *Collaboration*: Does not support collaboration features.
- *Live Code Integration*: Does not support live code integration.
- *Figure Zooming and Panning*: Does not provide built-in zoom or pan features.

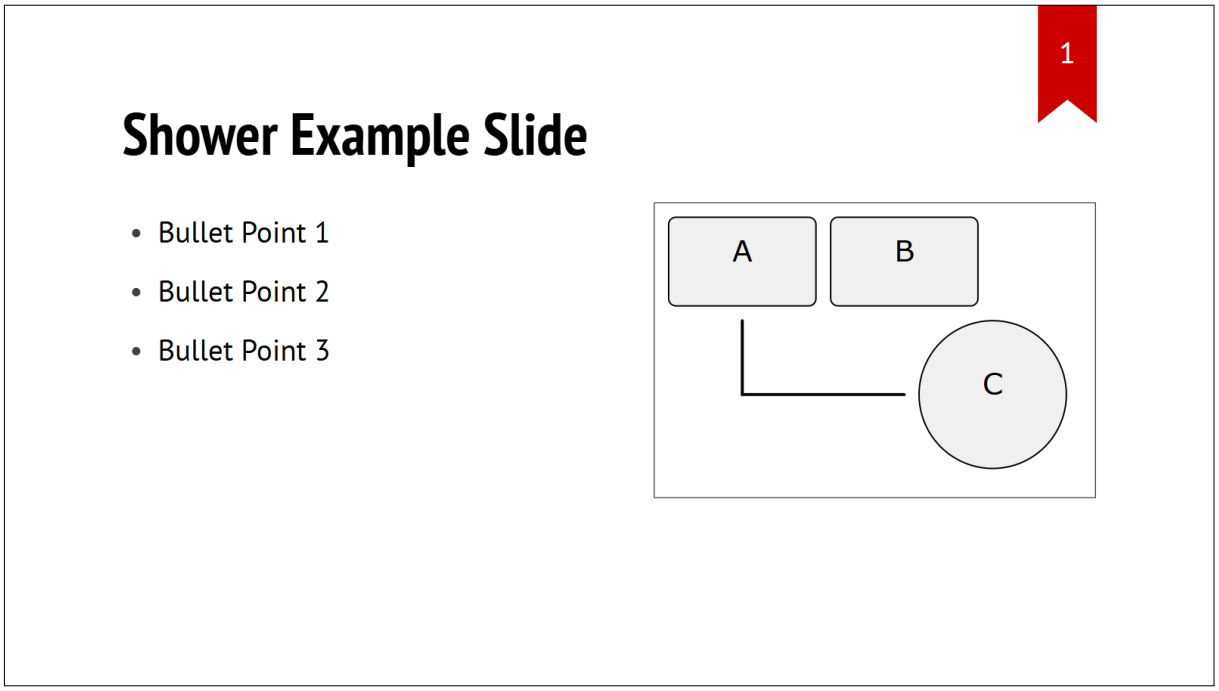


**Figure 3.1:** Shower: Example slide. The corresponding source code is shown in Listing 3.1.

```
<section class="slide">
  <h2>Shower Example Slide</h2>
  <div class="columns two">
    <ul>
        <li>Bullet Point 1</li>
        <li>Bullet Point 2</li>
        <li>Bullet Point 3</li>
    </ul>
    <div class="image">
        <img src="diagrams/diagram.svg" alt="Example Image">
    </div>
  </div>
</section>
```

**Listing 3.1:** Shower: Example slide source code. The resulting slide is shown in Figure 3.1.

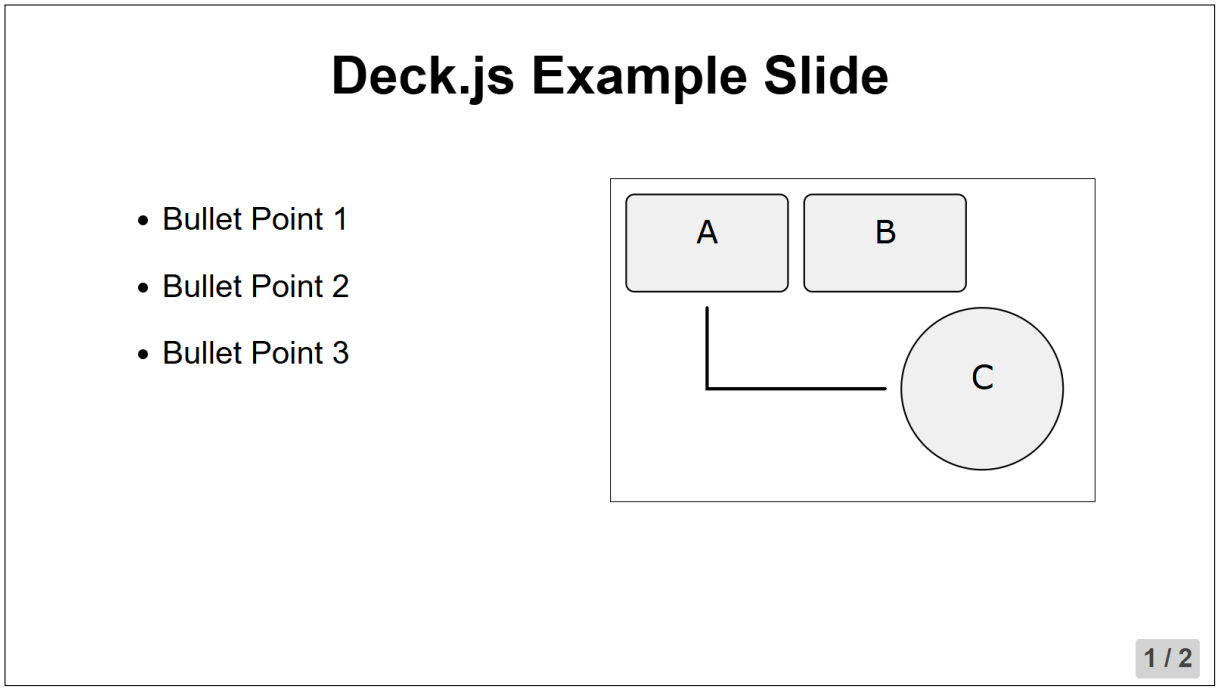


**Figure 3.2:** Deck.js: Example slide. The corresponding source code is shown in Listing 3.2.

## 3.2 Deck.js [2011-2016]

Deck.js is a flexible and extensible JavaScript library designed for creating HTML-based presentations, based on jQuery and Modernizr [Troughton 2016]. It provides templates and themes to build a simple standard slide deck.

The framework is not node-based and includes a file named boilerplate.html with all the necessary extensions included. This file serves as a starting point and users can immediately edit slides in this file and view them on a web browser. The basic functionality is provided by the deck.core module, which provides the functionality for creating and moving through a slide deck and defines various states for the slide deck that can be customised using CSS. Additional features are available through extensions and plugins. A simple example can be seen in Figure 3.2. The corresponding source code is shown in Listing 3.2. The key characteristics of Deck.js include:

+ *Nested Slides*: Embedding of slides within other slides.
+ *Markdown Support*: Markdown extension to create slides using Markdown syntax.
+ *Extensions and Plugins*:Deck.js has a vast variety of extensions and plugins to add advanced features.
+ *Interactive Navigation*: Presentations can be navigated using keyboard shortcuts, as well as touch gestures.
- *Export Options*: Does not support exporting presentations to PowerPoint or PDF formats.
- *Live Code Integration*: Does not support live code integration.
- *Figure Zooming and Panning*: Does not provide built-in zoom or pan features.
- *Limited Updates*: Last update was in 2016, which may indicate limited support or feature advancements.

```
<div class="slide">
  <h3 class="title">Deck.js Example SLide</h3>
  <div class="slide-content">
    <div class="bullets">
      <ul>
        <li>Bullet Point 1</li>
        <li>Bullet Point 2</li>
        <li>Bullet Point 3</li>
      </ul>
    </div>
    <div class="image">
      <img src="diagrams/diagram.svg" alt="Example Image">
    </div>
  </div>
</div>

<!-- Required JS files. -->
<script src="jquery.min.js"></script>
<script src="core/deck.core.js"></script>

<!-- Some Extensions for Deck.js -->
<script src="extensions/status/deck.status.js"></script>


<!-- Initialise the deck. -->
<script>
  $(function() { $.deck('.slide');});
</script>
```

**Listing 3.2:** Deck.js: Example slide source code. Slides are created with the class `slide`, usually either `<section class="slide">` or `<div class="slide">`. Deck.js offers various extensions, like the status extension, which displays the slide numbers on the example slide. The slide deck is initialised with the `deck('.slide')` function. The resulting slide is shown in Figure 3.2.

## 3.3 Reveal.js [2011-]

Reveal.js is an open-source JavaScript library for creating modern, web-based presentations [El Hattab 2024]. Like most other JavaScript-based slide deck tools, it allows users to create interactive slide decks using HTML, while enhancing them with CSS and JavaScript for smooth transitions and effects. Its basic features can be seen in Figure 3.3 and Listing 3.3. Advanced usages with different plugins can be seen in the demo version on GitHub [El Hattab 2024]. Additionally, slide decks using Reveal.js can be created and published via an online platform called Slides [El Hattab and Bossola 2018], which offers a fully-featured visual editor. They key characteristics of Reveal.js include:

+ *Vertical Slides*: Slides can be vertically nested up down as well as left right.
+ *Markdown Support*: Write content using inline or external Markdown.
+ *Code Syntax Highlighting*: Reveal.js supports code syntax highlighting with highlight.js.
+ *Auto-Animate*: Automatically animate matching elements across slides.
+ *Interactive Navigation*: Supports horizontal and vertical swiping to switch between slides.
+ *PDF Export*: Presentations can be exported to PDF.

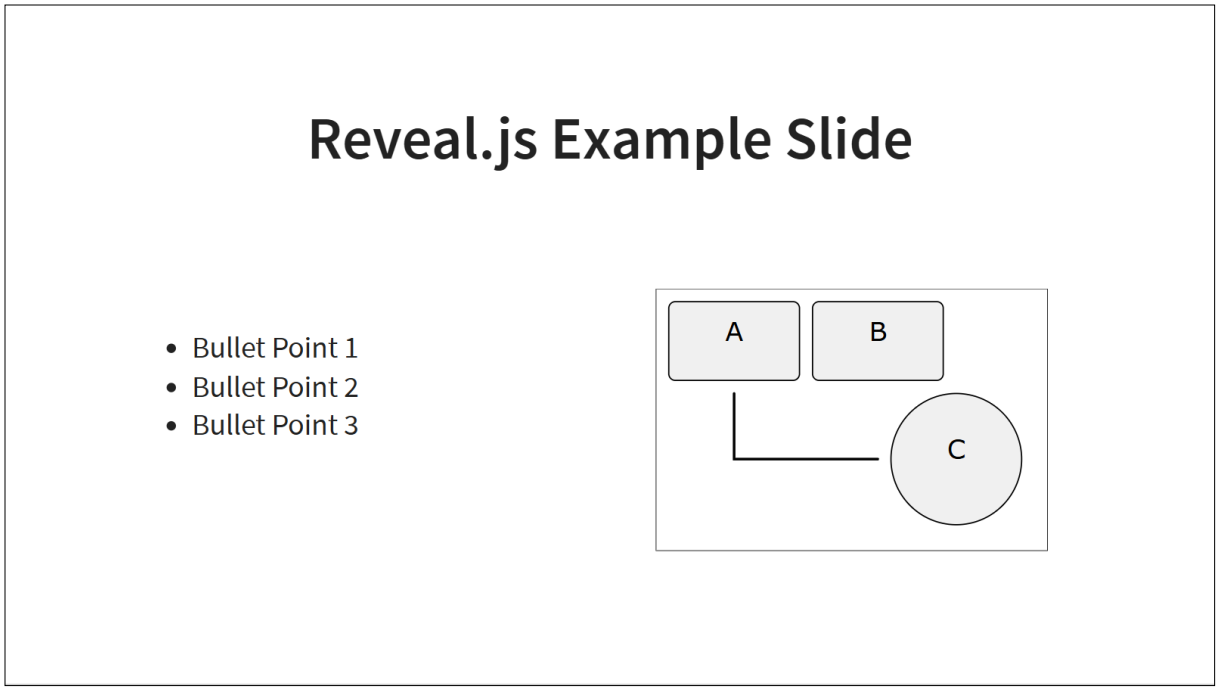


**Figure 3.3:** Reveal.js: Example slide. The corresponding source code is shown in Listing 3.3.

- *Collaboration*: Does not support collaboration features.
- *Live Code Integration*: Does not offer live code integration.
- *PowerPoint Export*: Not possible to export as PowerPoint.

## 3.4 impress.js [2011-]

impress.js is a presentation framework inspired by the idea behind Prezi [Halácsy et al. 2009]. It uses CSS3 transforms and transitions to create visually stunning and dynamic slide decks by positioning, rotating, and scaling them on an infinite canvas [Szopka and Ingo 2024]. This can be seen in Figure 3.4. To achieve this, each slide must include extra parameters for position, rotation, and scaling as HTML data attributes. The central position of a slide is defined by `data-x`, `data-y`, and `data-z`. Rotation is adjusted with `data-rotate-x`, `data-rotate-y`, and `data-rotate-z`. The scale of the slide is specified by `data-scale`, which defaults to 1. An example of a simple slide, where the slide is placed on the canvas can be seen in Figure 3.5, the corresponding source code is shown in Listing 3.4. The key characteristics of impress.js include:

+ *Flexible Slides*: The framework enables slides to be arranged in any order or path.
+ *3D Transformations and Transitions*: impress.js uses CSS3 transforms transitions across a 3D space.
+ *Media Embedding*: Easy embedding of multimedia elements such as images, videos, and audio.
+ *Dynamic Zooming*: Allows for zooming in and out on different slide elements.
+ *Interactive Navigation*: Presentations can be navigated using arrow keys or clicks to move and zoom between slides.
- *Collaboration*: Does not support collaboration features.
- *Live Code Integration*: Does not offer live code integration.

```
<body>
<div class="reveal">
  <div class="slides">
    <section>
      <h2 class ="title">Reveal.js Example Slide</h2>
        <div class="columns two">
          <ul>
            <li>Bullet Point 1</li>
            <li>Bullet Point 2</li>
            <li>Bullet Point 3</li>
          </ul>
          <div class="image">
            <img src="diagrams/diagram.svg" alt="Reveal.js">
          </div>
        </div>
    </section>
  </div>
</div>

<script src="dist/reveal.js"></script>
<script src="plugin/notes/notes.js"></script>
<script>
Reveal.initialize({
    hash: true,
});
</script>
</body>
```

**Listing 3.3:** Reveal.js: Example slide source code. Slides are handled in the `slides` class. Every `section` attribute creates a new slide. Calling the `Reveal.initialize()` function, creates the finished slide deck. The resulting slide is shown in Figure 3.3.

- *High Resource Usage*: Intensive use of CSS3 transforms can be resource-heavy.
- *Export Options*: Does not support exporting to PowerPoint or PDF formats.

## 3.5 Bespoke.js [2014-2020]

Bespoke.js is a highly modular presentation library designed for building slide decks in modern web applications [Dalgleish 2020]. The core library sets up the presentation structure, provides an intuitive control API, and manages events. Its modular design allows for extensive customisation, as additional features, functionality, and themes can be implemented through a vast selection of plugins offered on NPM [NPM 2024b]. Bespoke.js can be locally installed with the NPM command:

```
npm install -g generator-bespoke
```

Bespoke.js uses the Gulp task runner to automate and enhance workflows [Bublitz and Schoffstall 2024]. When the project is created, the gulp command:

```
gulp serve
```

can be used to run a preview server with LiveReload. A simple example slide can be seen in Figure 3.6 and the corresponding source code in Listing 3.5. The key characteristics of Bespoke.js include:

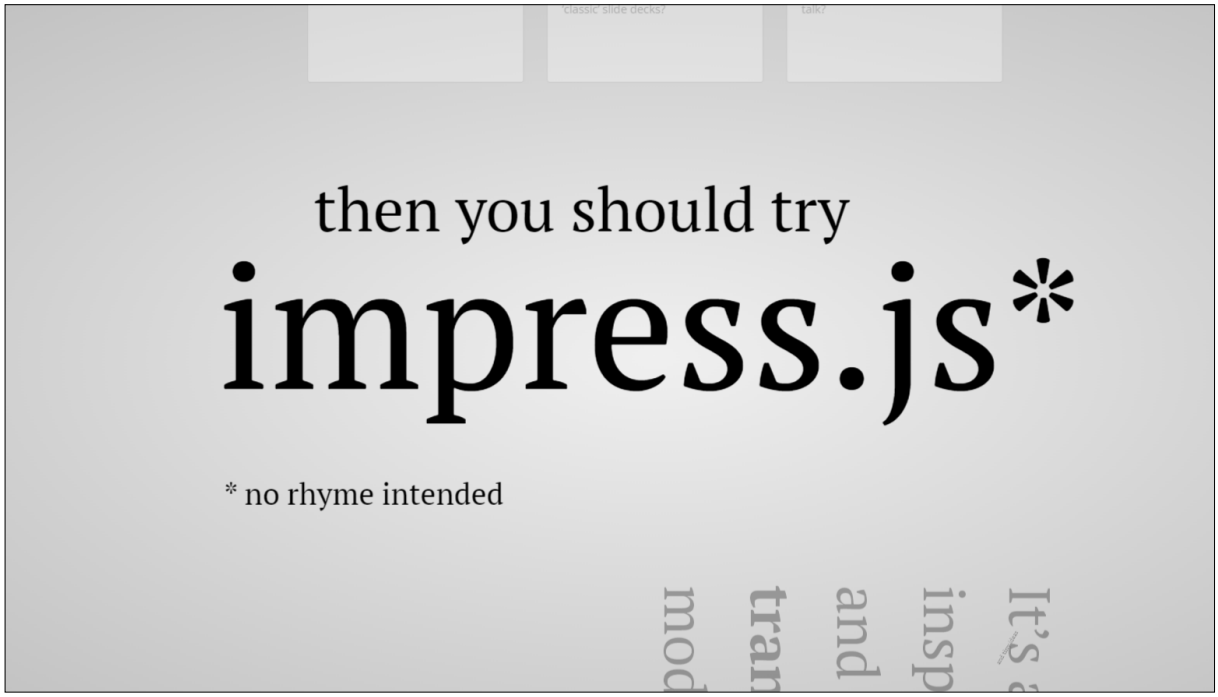


**Figure 3.4:** impress.js: Places all the slides on a canvas and then traverses it with a camera.

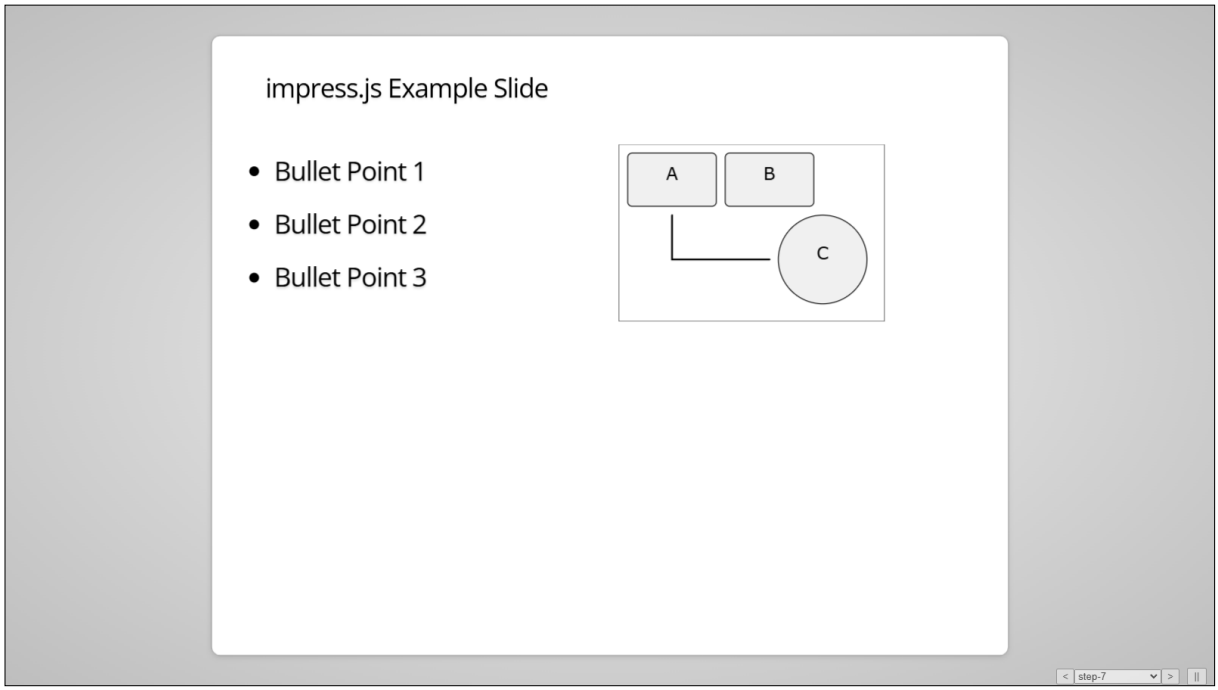


**Figure 3.5:** impress.js: Example slide. The corresponding source code is shown in Listing 3.4.

```
<div class="step slide" data-x="2000" data-y="-1500">
  <h1>impress.js Example Slide</h1>
  <br>
  <div class="container">
    <div class="column">
      <ul>
        <li>Bullet Point 1</li>
        <li>Bullet Point 2</li>
        <li>Bullet Point 3</li>
      </ul>
    </div>
    <div class="column">
      <img src="diagrams/diagram.svg" alt="Example image">
    </div>
  </div>
</div>
```

**Listing 3.4:** impress.js: Example slide source code. The resulting slide is positioned on the impress.js canvas with the `data-x` and `data-y` attributes, as shown in Figure 3.5.

+ *Modular Architecture:* Allows for extensive customisation and extension through a variety of plugins.
+ *Lightweight Design:* Small core library (1KB minified and gzipped), ensures fast load times and efficient performance.
+ *Intuitive Control API:* Straightforward API for controlling and managing presentations.
+ *Plugin Network:* Wide range of plugins for additional functionalities and themes, available through NPM.
+ *Interactive Navigation:* Keyboard and touch interaction for slide navigation.
+ *PDF Export*: Presentations can be exported to PDF.
- *Collaboration*: Does not support collaboration features.
- *Live Code Integration*: Does not offer live code integration.
- *Limited Native Features*: As a modular framework, Bespoke.js does not include many advanced features natively.

## 3.6 Inspire.js [2018-]

Inspire.js, formerly knows as CSS-based SlideShow System (CSSS), is a JavaScript-based slide tool with a plugin-based architecture, allowing users to easily extend its functionality [Verou 2024]. Slides are defined by adding the `slide` class to any block-level element, as shown in Figure 3.7 and Listing 3.6. The key characteristics of Inspire.js include:

+ *Repeated Slides*: The same slide can be easily repeated multiple times in a presentation, based on its id.
+ *Supports Annotated Videos*: Embedding of videos with added annotations.

```
<section>
  <h2>Bespoke.js Example Slide</h2>
  <div class="slide-content">
   <div class="bullets">
    <ul>
      <li>Bullet Point 1</li>
      <li>Bullet Point 2</li>
      <li>Bullet Point 3</li>
    </ul>
   </div>
   <div class="image">
     <img src="images/diagram.svg" alt="Inspire.js">
   </div>
  </div>
</section>

<style>
.slide-content {
  display: flex;
    justify-content: space-between;
    padding-top: 2rem;
    width: 80%;
  }
  .bullets {
    width: 50%;
    display: flex;
    align-items: flex-start;
    padding-top:  1rem;
    padding-left: 0;
  }
  .image {
    width: 50%;
    display: flex;
    align-items: center;
    justify-content: center;
  }
</style>
```

**Listing 3.5:** Bespoke.js: Example slide source code. The slide content is placed inside a `section` element. The resulting slide is shown in Figure 3.6.

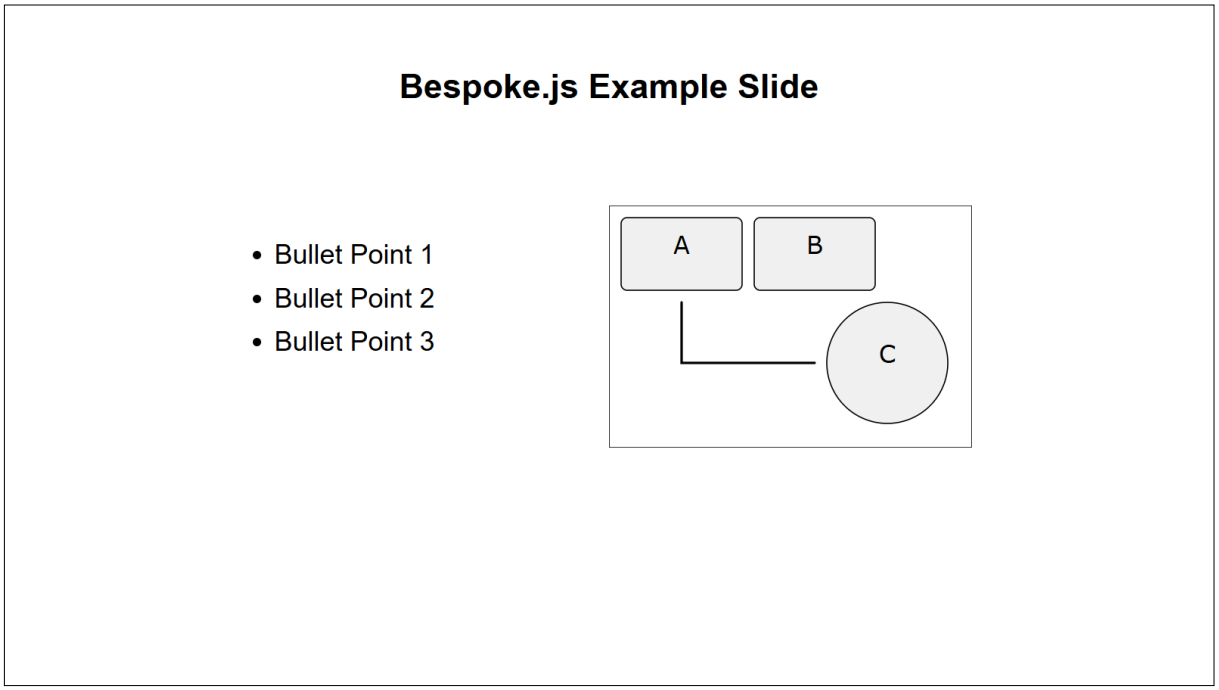


**Figure 3.6:** Bespoke.js: Example slide. The corresponding source code is shown in Listing 3.5.

+ *Incremental Display of Content*: Slide elements can be displayed progressively.
+ *Code Syntax Highlighting*: Code syntax can be highlighted with integrated PrismJS [Verou 2022].
+ *Presenter Mode*: Plugins for presenter notes and views.
- *Collaboration*: Does not support collaboration features.
- *Live Code Integration*: Does not offer live code integration.
- *Export Options*: Does not support exporting to formats like PowerPoint or PDF.

## 3.7 Comparison of JavaScript-Based Slide Decks

JavaScript-based slide deck tools are compared in Table 3.1. All of the JavaScript-based slide deck tools support the native input of SVG graphics and slide numbering but they all lack stand-alone collaborative editing capabilities.

Each tool offers customisation options for bullet indentation and spacing in master templates. Exporting presentations as PDFs is possible with Bespoke.js, Reveal.js, and Shower. Furthermore, since all of these tools are built on HTML, they all support exporting content in that format. However, none of them provide the option to directly export to PowerPoint, which could be a limitation for users needing that specific format.

The update frequency for these slide deck tools varies, with Inspire.js and Reveal.js having recent updates, while Deck.js has not been updated since 2016. In terms of popularity, Reveal.js stands out with 67.4 thousand GitHub stars, followed by impress.js with 37.6 thousand GitHub stars, indicating their widespread use and an active community.

Navigation features such as margin tap and swipe gestures to navigate slides are supported by all tools. A presenter mode, which provides a second window for speaker notes, is available across all slide deck tools. Additionally, all the slide deck tools are distributed under the MIT license and are open-source.

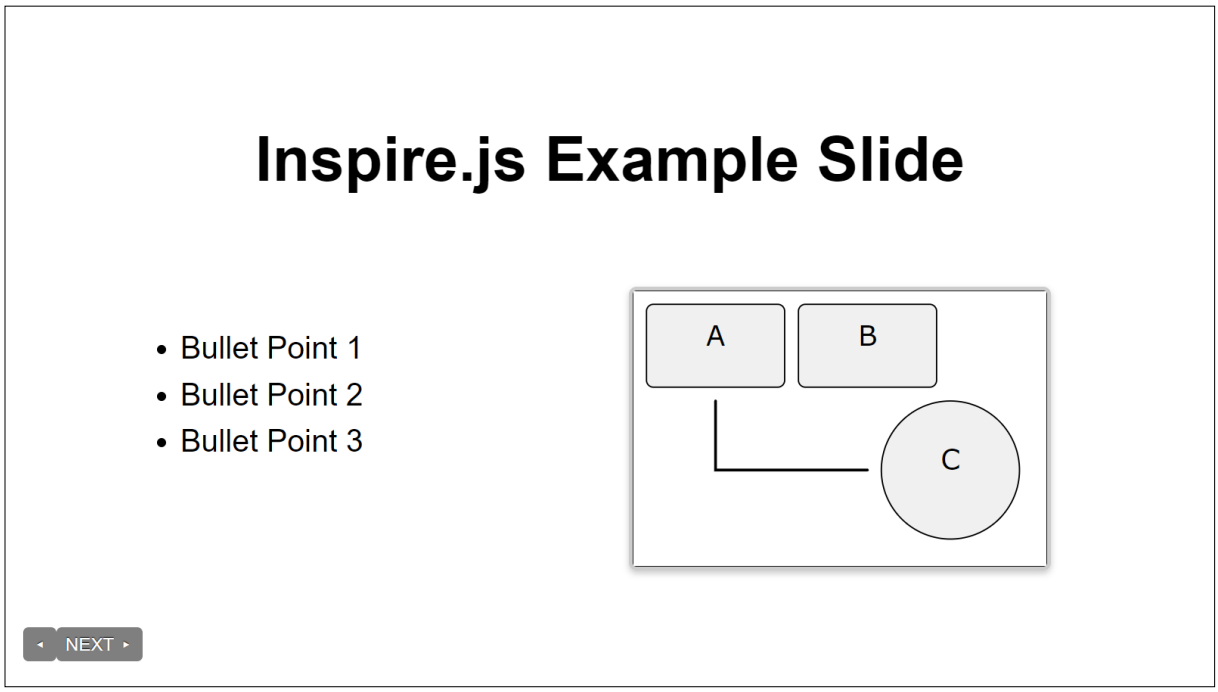


**Figure 3.7:** Inspire.js: Example slide. The corresponding source code is shown in Listing 3.6.

```
<article class="slide" id="example-slide">
<h1>Inspire.js Example Slide</h1>
<div class="slide-content">
  <div class="bullets">
    <ul>
      <li>Bullet Point 1</li>
      <li>Bullet Point 2</li>
      <li>Bullet Point 3</li>
    </ul>
  </div>
  <div class="image">
    <img src="./diagrams/diagram.svg" alt="Inspire.js">
  </div>
</div>
</article>
```

**Listing 3.6:** Inspire.js: Example slide source code. The slide content is placed inside an `<article class="slide">` element. The resulting slide is shown in Figure 3.7.

| | Shower | Reveal.js | Deck.js | impress.js | Bespoke.js | Inspire.js |
|---|---|---|---|---|---|---|
| Collaborative: | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| SVG Inclusion: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Globally Change Bullet Spacing: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Globally Change Bullet Indentation: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Export as PDF: | ✓ | ✓ | ✗ | ✗ | ✓ | ✗ |
| Export as HTML: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Export as PowerPoint: | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| Slide Numbering: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Live Code Integration: | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| Open-Source: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Licence: | MIT | MIT | MIT | MIT | MIT | MIT |
| First Release: | 2010-10-25 | 2011-06-07 | 2011-06-24 | 2011-12-28 | 2012-12-13 | 2018-09 |
| Last Update: | 2024-10-25 | 2024-05-15 | 2016-05-04 | 2024-04-26 | 2020-09-08 | 2024-06-27 |
| Popularity (GitHub stars): | 4.8k | 67.4k | 5.4k | 37.6k | 4.7k | 1.7k |
| Swipe Navigation: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Margin Navigation: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Figure Zooming and Panning: | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ |
| Presenter Mode: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

**Table 3.1:** Comparison of JavaScript-based slide deck tools.

# Chapter 4

# Hosted Slide Decks

Hosted slide deck tools are online tools, that allow users to create, manage, and share presentation slides over the internet without needing to install software locally. These tools offer a range of modern features such as AI-powered tools and built-in drag and drop editors. Presentations can be shared via a simple link, allowing for easy access and real-time collaboration without downloading files. Many tools also offer embedding options for websites and built-in integrations with productivity suites such as Microsoft 365 and Google Workspace. Most hosted slide deck tools have free versions with limited features, but offer subscription models for different advanced features such as AI-designs and PDF export.

This chapter describes six popular hosted tools, which can be used to create online presentations: Google Slides, Zoho Show, Prezi, Visme, Mentimeter, and Microsoft Sway. Section 4.7, at the end of the chapter, compares the tools.

## 4.1 Google Slides [2006-]

Google Slides is a cloud-based software and part of the free, web-based Google Docs suite offered by Google [Google 2024]. It allows users to collaborate on presentations in real-time, and present slide decks online. Google Slides can easily be exported as a PowerPoint presentation or PDF. A simple Google Slides example slide can be seen in Figure 4.1.

The key characteristics of Google Slides include:

+ *Google Workspace Integration*: Integrates seamlessly with Google Drive, Sheets, and Docs.
+ *Offline Editing*: Offline presentation editing with automatic syncing when reconnected to the internet.
+ *Version History*: Track and restore previous versions of presentations.
+ *Export Options*: Allows exporting presentations to PowerPoint and PDF formats.
+ *Collaborative*: Supports real-time collaborative editing.
+ *Swipe Navigation*: Slides can be navigated with swiping gestures.
- *SVG Inclusion*: Not possible to include SVG vector graphics.
- *Live Code Integration*: Does not support integration of live code.
- *Figure Zooming and Panning*: Does not provide built-in zoom or pan features.
- *Global Changes to Bullet Spacing*: Not possible to change bullet spacing in master template.

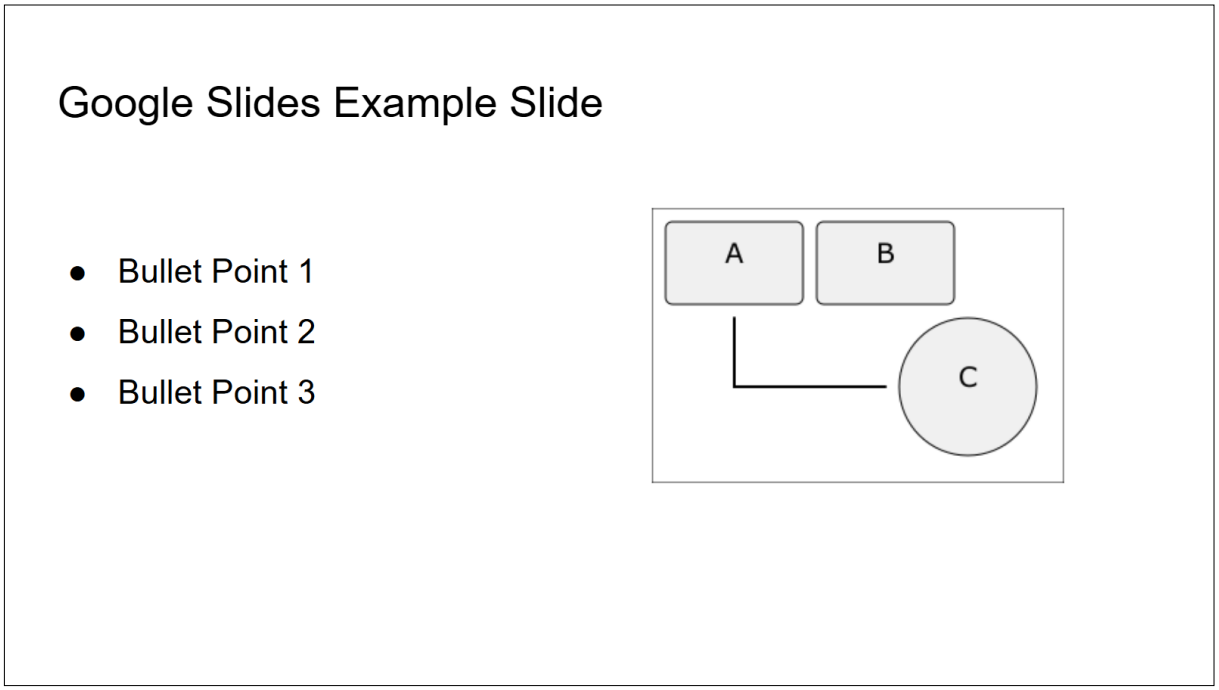


**Figure 4.1:** Google Slides: Example slide with an PNG image on the right (Google Slides does not support SVG).

## 4.2 Zoho Show [2006-]

Zoho Show is a collaborative presentation tool for modern teams [Zoho 2024]. It offers a user-friendly platform for creating, editing, and delivering professional presentations. Zoho Show is mostly known for its wide range of templates, themes, and design tools, as can be seen in Figure 4.2. Users can easily customise these themes and templates to their needs. As part of the Zoho suite, it integrates seamlessly with other Zoho applications. Zoho Show offers many features and it shares similarities with Visme, like the very user-friendly drag-and-drop editor. Due to its many features and huge range of templates and themes, it can be overwhelming and hard to navigate. Certain features, such as offline mode, advanced real-time collaboration, enhanced file sharing, custom branding, and additional integrations, are only available with paid plans. A simple example slide can be seen in Figure 4.3.

The key characteristics of Zoho Show include:

+ *Collaboration*: Supports real-time collaboration.
+ *Integration*: Easy integration with other Zoho applications, such as Zoho CRM and Zoho Mail.
+ *Rich Template Library*: Wide selection of customisable templates.
+ *User-Friendly Interface*: User-friendly interface, including an easy-to-use drag-and-drop editor.
+ *Data Visualisation Tools*: Includes a variety of charts, graphs, and data widgets for data visualisation.
- *SVG Inclusion*: Does not support the inclusion of vector graphics (SVG).
- *Live Code Integration*: Does not support the integration of live code within presentations.
- *Figure Zooming and Panning*: Does not provide built-in zoom or pan features for presentation elements.
- *Advanced Features*: Many advanced features and templates are restricted to paid users.

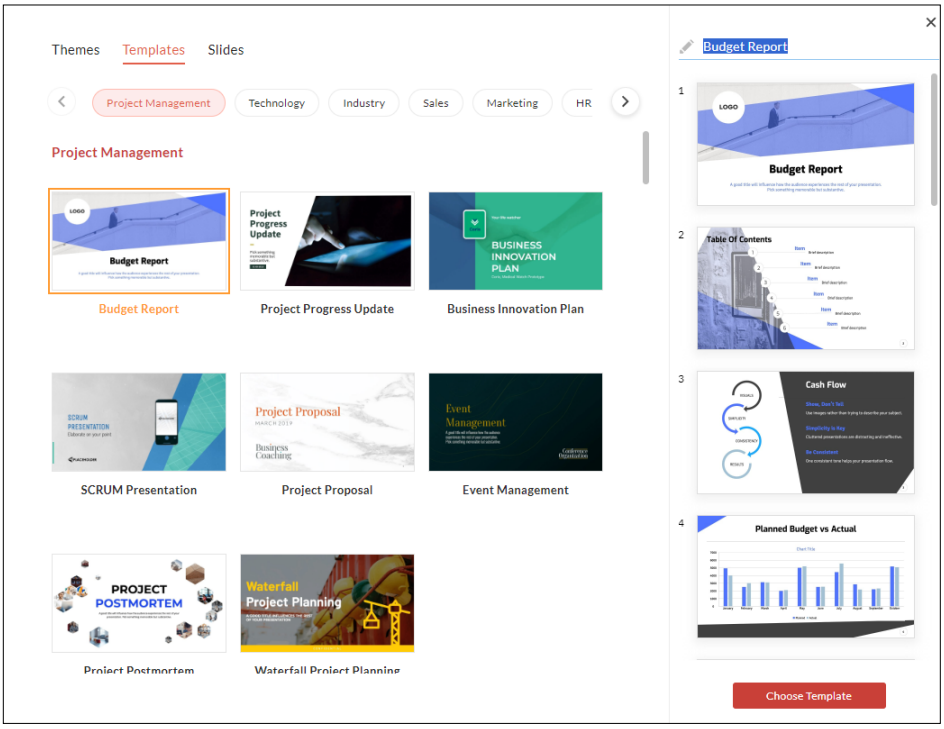


**Figure 4.2:** Zoho Show: Templates with almost fully finished slides to build up on.

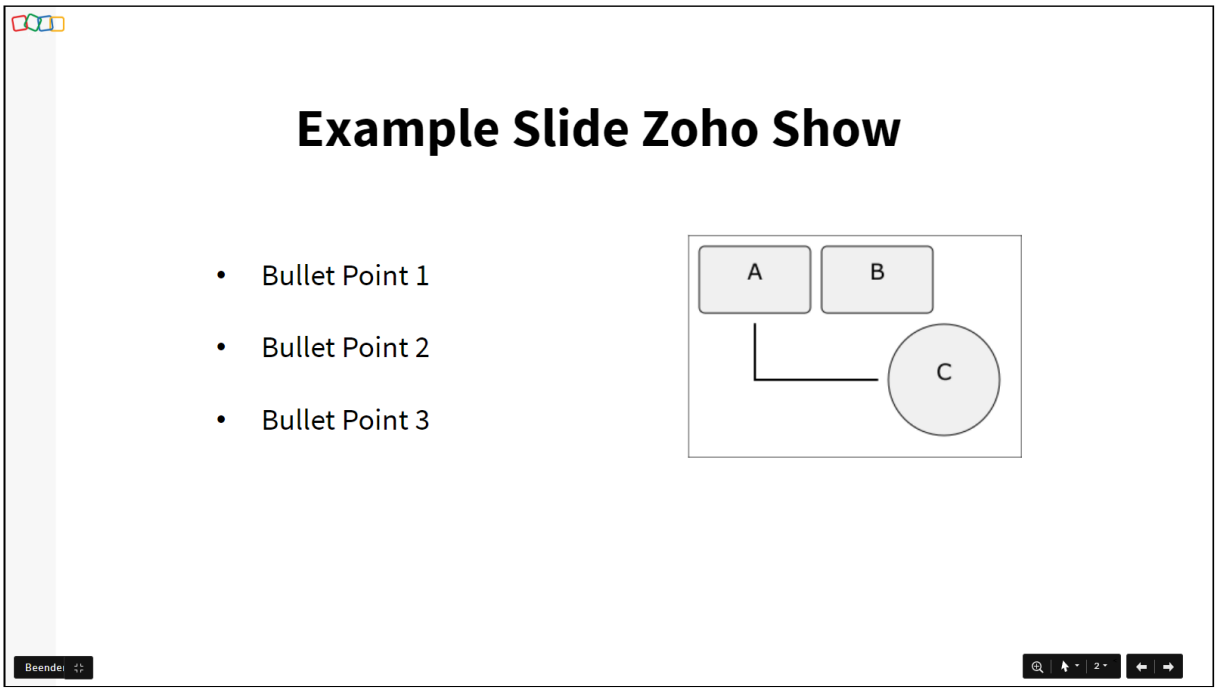


**Figure 4.3:** Zoho Show: Example slide with an PNG image on the right. Zoho Show does not support import of SVG vector graphics.

## 4.3 Prezi [2009-]

Prezi is an engaging web-based presentation tool that focuses on animations and offers a map-like overview allowing users to navigate between topics and slides, and zoom in and out for slide details [Halácsy et al. 2009]. With Prezi, users can create visually stunning slide decks, where the slides are placed on a very large canvas, similar to impress.js. An example slide can be seen in Figure 4.4. The corresponding canvas the slide is placed on is shown in Figure 4.5. Prezi offers a tiered pricing model with various plans designed for different user needs. The plans include a free basic version with limited features, and several paid subscriptions, such as individual, business, and education.

The key characteristics of Prezi include:

+ *Collaboration:* Real-time collaboration, allowing multiple users to edit and design presentations simultaneously (paid plans).
+ *Embedded Content:* Images, videos, charts, and other multimedia elements can be directly embedded onto the canvas.
+ *Integration with Popular Tools:* Seamless integration with tools such as Zoom, Microsoft Teams, and Slack.
+ *Zoomable Canvas:* Interface allows users to move freely across a large canvas, zooming in and out on specific elements.
+ *Non-Linear Navigation:* Presentations do not have to follow a fixed order, allowing users to move freely around the canvas
- *Live Code Integration*: Does not support integration of live code.
- *Global Changes to Bullet Spacing and Indentation*: Limited formatting options for bullet spacing and indentation in master template.
- *SVG Inclusion*: Not possible to include SVG vector graphics.
- *Export Options*: Not possible to export presentation as PowerPoint or HTML formats.

## 4.4 Visme [2013-]

Visme is an all-in-one visual content creation tool that allows users to design a wide range of material, such as presentations, infographics, reports, social media graphics, and more [Taei 2024a]. It simplifies content creation with its user-friendly interface, while offering a very modern drag-and-drop tool, as shown in Figure 4.6. Although Visme offers many tools, most of the features are only available with a paid subscription, such as real-time collaboration, expanded download options, advanced customisation tools, brand kits, analytics, and offline presentation [Taei 2024b]. An example slide can be seen in Figure 4.7

The key characteristics of Visme include:

+ *Collaboration*: Supports real-time collaboration.
+ *SVG Inclusion*: Possible to include SVG vector graphics.
+ *Template Library*: Wide range of customisable templates for different content types.
+ *AI-Powered Tools*: AI-tools to offer design suggestions based on the users input.
+ *Digital Whiteboard*: Collaborative digital whiteboard that multiple users can edit in real-time.
+ *Extensive Asset Library*: Vast library of free assets such as images, icons, fonts, and other design

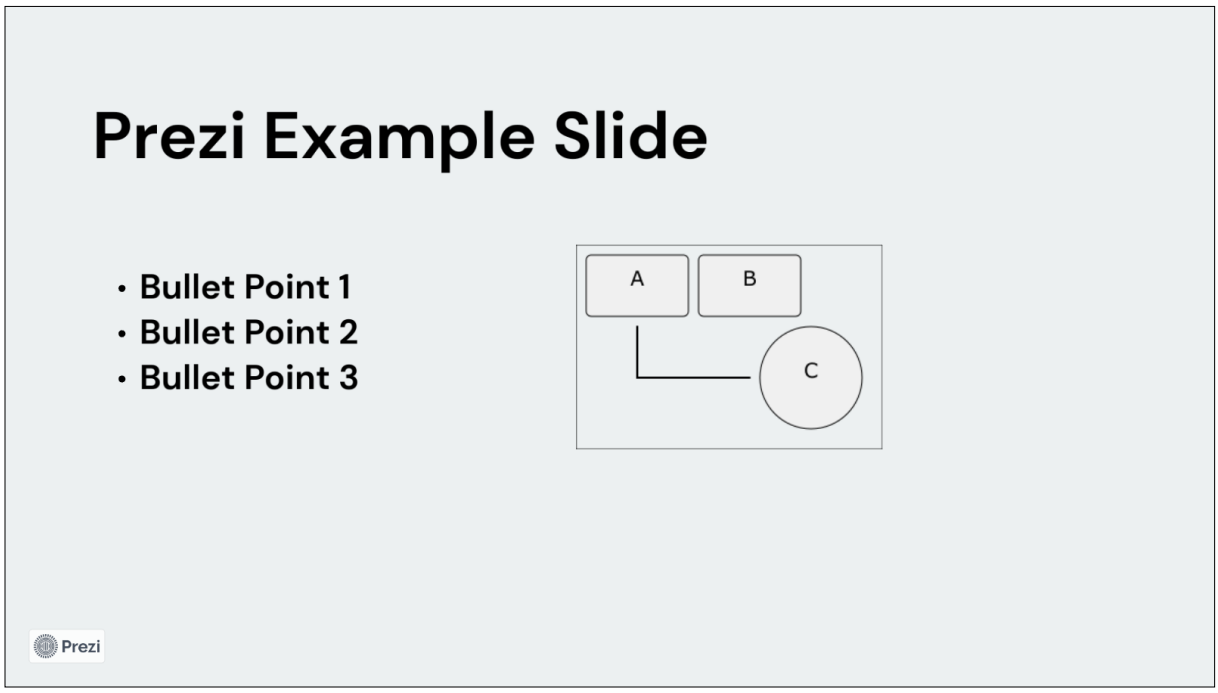


**Figure 4.4:** Prezi: Example slide. The slide is placed on a canvas. The colour palette of the slide and canvas can only be fully customised with a paid subscription.

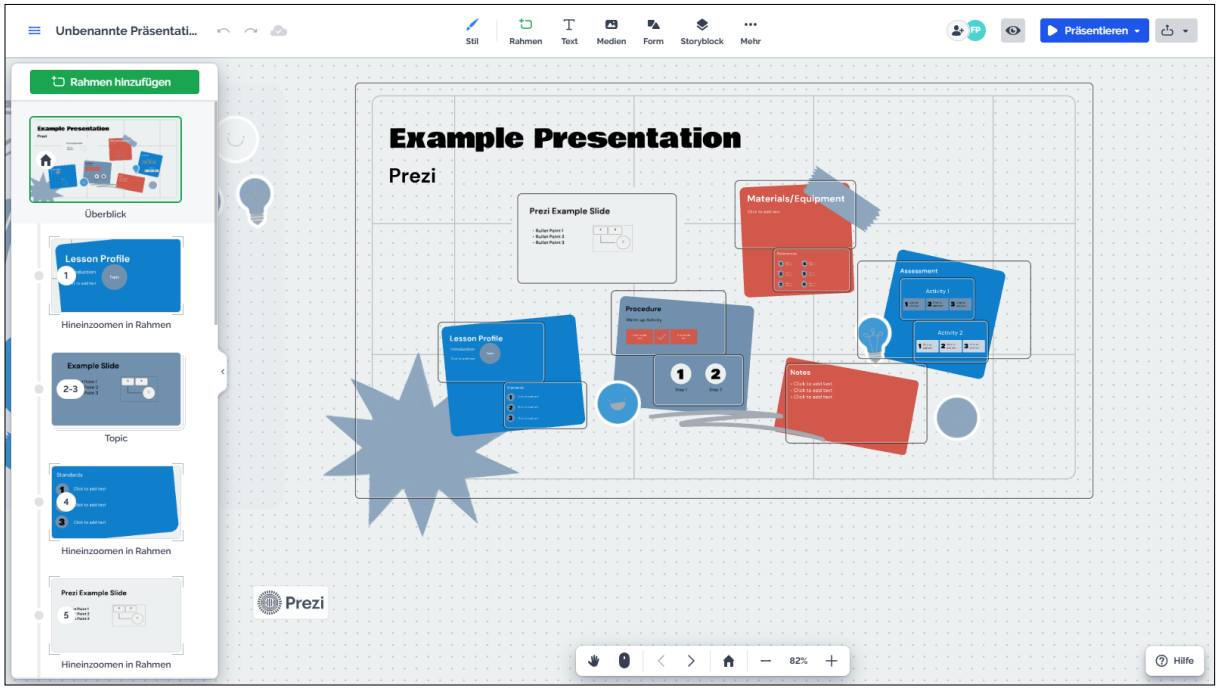


**Figure 4.5:** Prezi: Canvas with presentation slides. When presenting, the author can zoom in and out of the canvas to reach specific slides.

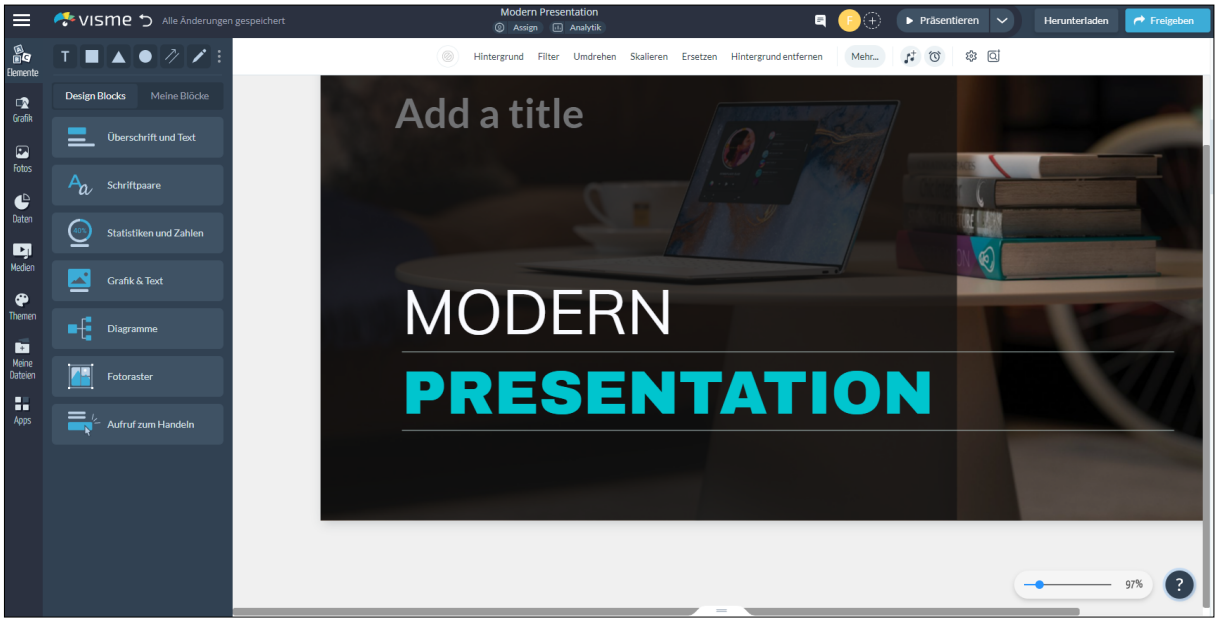


**Figure 4.6:** Visme: User friendly drag-and-drop editor.

elements.

+ *Built-In Analytics*: Analytics tools that provide insights into how content is performing.

+ *Export Options*: Presentations can be exported to PowerPoint, PDF and HTML formats.

- *Integration of Live Code*: Does not support integration of live code.

- *Global Changes to Bullet Spacing and Indentation*: Limited formatting options for bullet spacing and indentation in master template.

- *Advanced Features*: Many advanced features and templates are restricted to paid users.

## 4.5 Mentimeter [2014-]

Mentimeter is an interactive presentation tool that allows presenters to engage their audience in real-time [Warström et al. 2012]. Unlike most other slide deck tools, in Mentimeter, users can create various types of interactive content such as polls, quizzes, word clouds, and Q&A sessions. Audience members can participate using their smartphones or other devices by accessing a unique code or link provided by the presenter. Mentimeter provides visualisations of the responses in real-time, making presentations more dynamic and engaging. However, some of its advanced features such as unlimited slides, advanced audience interaction tools (such as moderation of Q&A), customised branding options and the ability to export data to Excel, are only available with paid plans. It is often used in classrooms, meetings, workshops, and conferences to gather feedback and enhance audience involvement, mostly with the help of polls. A simple example slide can be seen in Figure 4.8. To also show one of Mentimeters most used features, a test poll was created, as shown in Figure 4.9.

The key characteristics of Mentimeter include:

+ *Accessibility*: Participants can join and interact from any device with an internet connection.

+ *Real-Time Results*: Audience responses are displayed live in real-time.

+ *Integration*: Seamless integration with PowerPoint, Google Slides, and other presentation software.

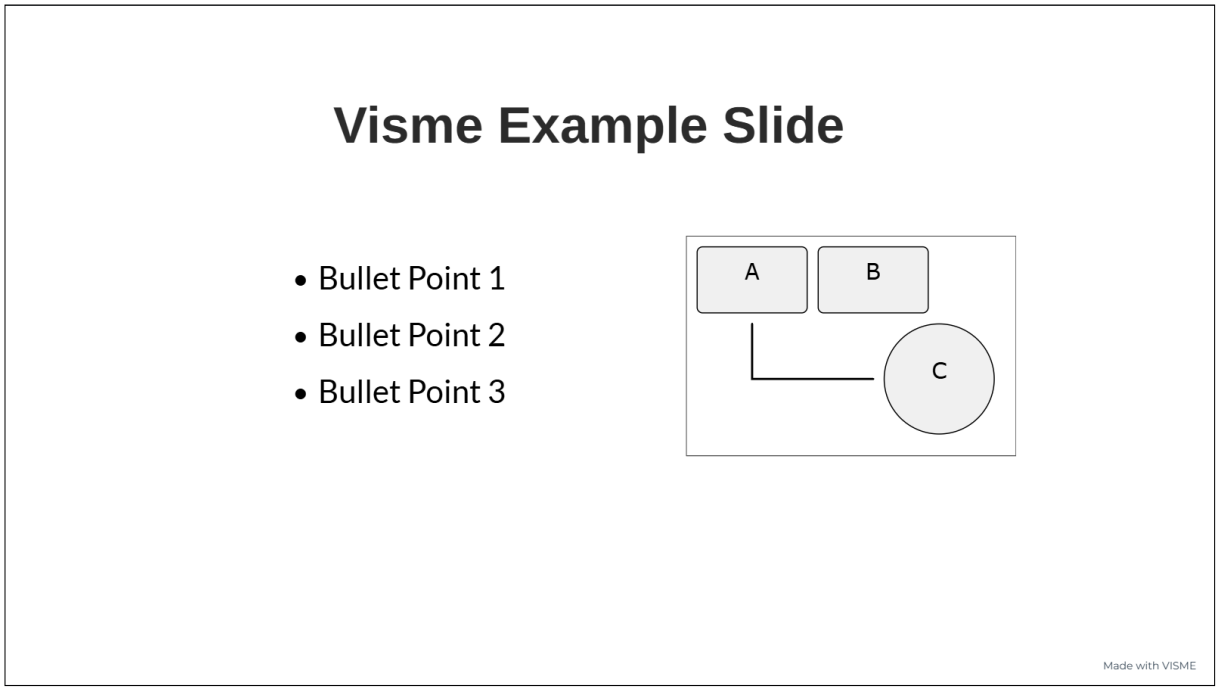


**Figure 4.7:** Visme: Example slide in its predefined presentation window. The slide can only be fully seen or downloaded with a paid subscription model.

+ *Data Visualization*: Responses can be visualised as charts, or graphs.
+ *Reporting and Analytics*: Detailed analytic reports and data export options.
+ *SVG Inclusion*: Supports SVG vector graphics.
- *Integration of Live Code*: Does not support integration of live code.
- *Figure Zooming and Panning*: Does not provide built-in zoom or pan features.
- *Export Options*: Not possible to export presentation as PowerPoint or HTML formats.
- *Advanced Features*: Many advanced features are restricted to paid plans.
- *Slide Numbering*: Does not support slide numbering.

## 4.6 Microsoft Sway [2015-]

Microsoft Sway is an app from Microsoft Office that makes it easy to create and share interactive reports, personal stories, presentations, and more [Microsoft 2024]. Content in Microsoft Sway is organised as a storyline, where users can add text, images and multimedia elements. Its built-in design engine automatically formats the content. Users can switch styles and designs, but are limited in terms options. Microsoft Sway automatically optimises presentations for different devices, ensuring a refined look without the need for design expertise. As part of the Microsoft Office 365, Microsoft Sway can be easily integrated in other Microsoft services. An example slide with the standard pre-defined Microsoft Sway design can be seen in Figure 4.10.

The key characteristics of Microsoft Sway include:

+ *Collaborative*: Multiple users can work on the same Sway project, with real-time updates.
+ *Multimedia Embedding*: Easy embedding of multimedia elements.

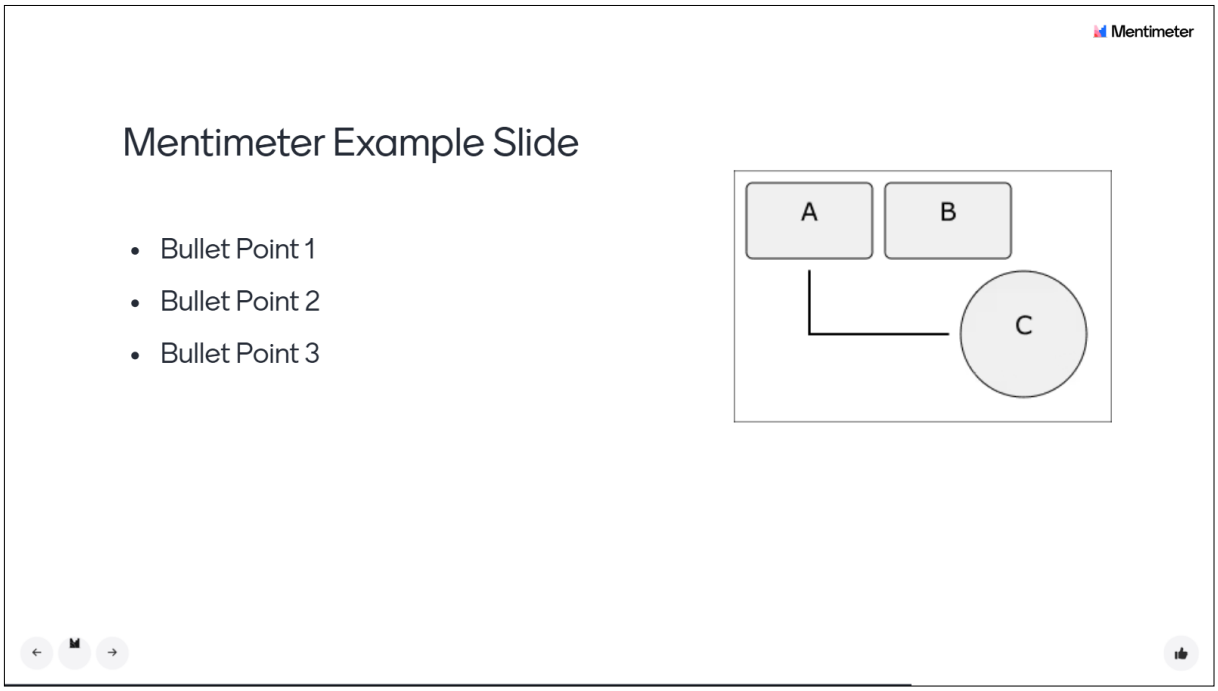


**Figure 4.8:** Mentimeter: Example slide with an SVG vector graphic on the right.

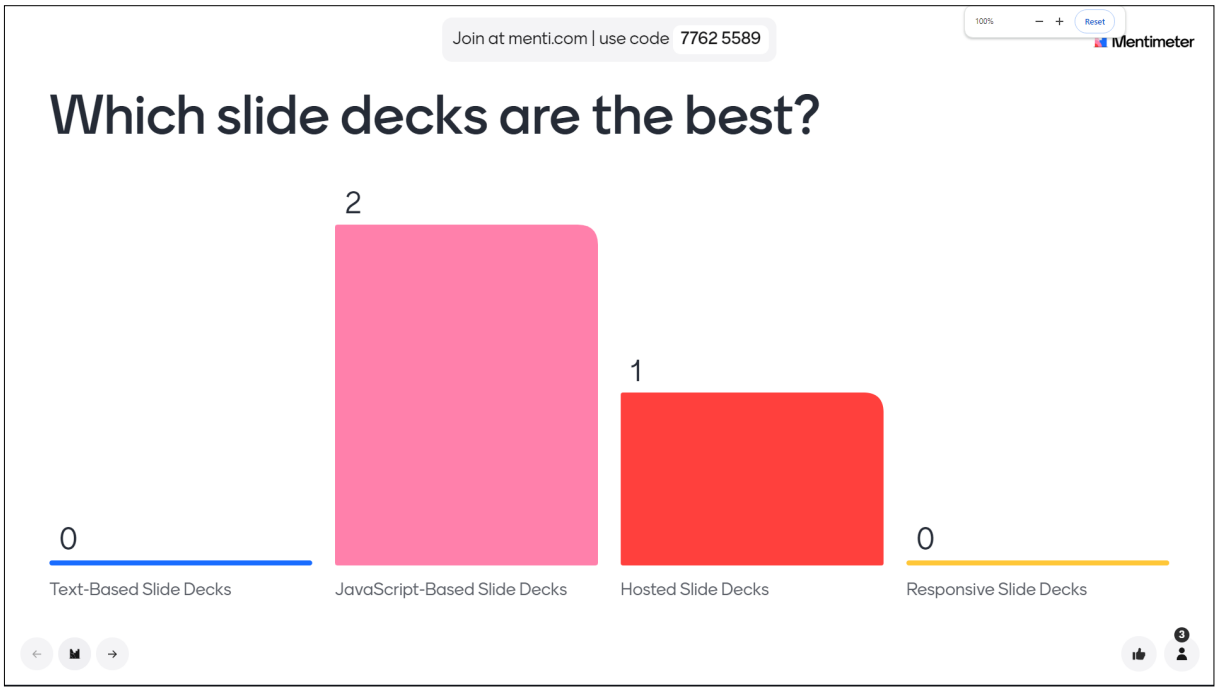


**Figure 4.9:** Mentimeter: Example poll about which slide decks are the best, based on an example user voting.

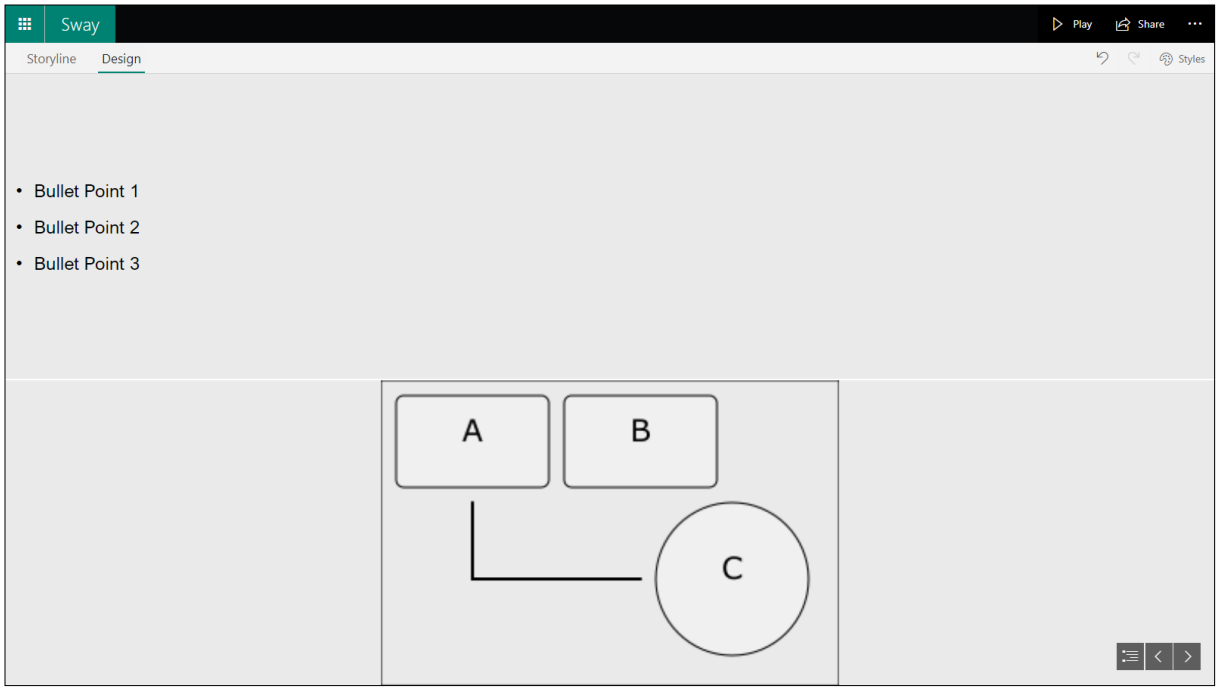


**Figure 4.10:** Microsoft Sway: Example slide with PNG image. Microsoft Sway does not support import of SVG vector graphics. The styling options in Microsoft Sway are intentionally limited, ensuring ease of use, cross-device compatibility, and streamlined visual storytelling.

+ *Integration with Microsoft 365*: Seamless integration with other Microsoft 365 tools.
+ *Design Suggestions and Usability*: Intelligent design suggestions and user-friendly drag-and-drop interface.
+ *Analytics*: Provides viewer analytics to track engagement with shared sway presentations.
+ *Advanced Image Viewer*: Includes zoom and pan features for enhanced image interaction.
- *Customisation Limitations*: Limited ability to customise design beyond pre-set templates.
- *Export Options*: Does not support exporting to PowerPoint or HTML formats.
- *Global Changes to Bullet Spacing and Indentation*: Not possible to change bullet spacing and indentation in master template.
- *Slide Numbering*: Does not support slide numbering.

## 4.7 Comparison of Hosted Slide Decks

While many hosted slide deck tools offer similar core features, notable differences arise across specific functionalities, as summarised in Table 4.1. All tools support real-time collaboration, allowing multiple users to work on presentations simultaneously. However, only Visme and Mentimeter support the native import of vector graphics (SVG), making them ideal for design-heavy presentations.

Google Slides and Zoho Show allow changes to bullet indentation within the master template, but Visme, Prezi, Microsoft Sway, and Mentimeter do not support bullet indentation or spacing adjustments globally. When it comes to export options, all tools support PDF export, but only Visme, and Zoho Show offer the ability to export as HTML. PowerPoint export is supported by Google Slides, Visme and Zoho Show. Slide numbering is available across most tools, except for Microsoft Sway and Mentimeter, which

| | Google Slides | Zoho Show | Prezi | Visme | Mentimeter | Microsoft Sway |
|---|---|---|---|---|---|---|
| Collaborative: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| SVG Inclusion: | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ |
| Globally Change Bullet Spacing: | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ |
| Globally Change Bullet Indentation: | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ |
| Export as PDF: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Export as HTML: | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ |
| Export as PowerPoint: | ✓ | ✓ | ✗ | ✓ | ✗ | ✗ |
| Slide Numbering: | ✓ | ✓ | ✓ | ✓ | ✗ | ✗ |
| Live Code Integration: | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| Open-Source: | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| Licence: | Proprietary | Proprietary | Proprietary | Proprietary | Proprietary | Proprietary |
| First Release: | 2006-03-09 | 2006 | 2009-04-05 | 2013 | 2014 | 2014-08 |
| Last Update: | Frequent | Frequent | Frequent | Frequent | Frequent | Frequent |
| Popularity (Usage): | High | Medium | High | Medium | Medium | Medium |
| Swipe Navigation: | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Margin Navigation: | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ |
| Figure Zooming and Panning: | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ |
| Presenter Mode: | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ |

**Table 4.1:** Comparison of hosted slide deck tools.

do not follow traditional slide structures. While Prezi and Microsoft Sway offer zoom and pan features for interactive presentations, other tools lack this functionality.

All tools support swipe gestures for moving between slides. However, only Google Slides and Zoho Show also offer margin tapping for navigation. Google Slides, Prezi, Visme, Zoho Show, and Mentimeter support a dedicated presenter mode. Lastly, while all the tools are proprietary and are updated frequently, their popularity varies. Google Slides and Prezi are the most widely used, while Microsoft Sway, Visme, Zoho Show, and Mentimeter have a more moderate user base.

# Chapter 5

# Responsive Slide Decks

Responsive web design is a modern approach to designing web sites and web applications which adapt to the characteristics of the end user's device. The same content is served to every device from the same URL, but its internal logic supports different layouts and input modalities. Responsive slide decks are web-based slide decks which can adapt themselves to various display devices, by responding to different screen widths, and supporting mouse, touch, and keyboard input.

## 5.1 Rslidy [2019-]

Rslidy is a modern, lightweight, and responsive presentation tool designed to run directly in web browsers using HTML5, CSS3, and TypeScript [Rslidy 2024; Hipp 2021]. Building on the idea of Slidy2 [Raggett 2006], Rslidy improves the original concept by enhancing responsiveness with the usage of web design techniques such as CSS3 Media Queries and containers, to various browsers and different resolution devices. It also adds a range of new features, while being self-contained, which means, no additional dependencies are needed. Slides are created within `<section>` or `<div class="slide">` elements, and are presented one at a time with transitions. An example slide can be seen in Figure 5.1 and the corresponding Listing 5.1.

The key characteristics of Rslidy include:

+ *Standard HTML5 Slides*: Slides can be created with standard HTML5 elements.
+ *Responsive Design*: Slides can be made responsive on all modern browsers and mobile devices.
+ *Interactive Navigation*: Slides can be navigated using keyboard controls, as well as touch, tilt, shake, and swipe gestures.
+ *Live Code Integration*: Offers embedding of live code.
+ *Figure Zooming and Panning*: Includes zoom and pan features for enhanced image interaction.
+ *SVG Support*: SVG vector graphics can be included into presentations.
- *Collaboration*: Does not support collaboration features.
- *Export as PowerPoint*: Not possible to export the presentation as PowerPoint.

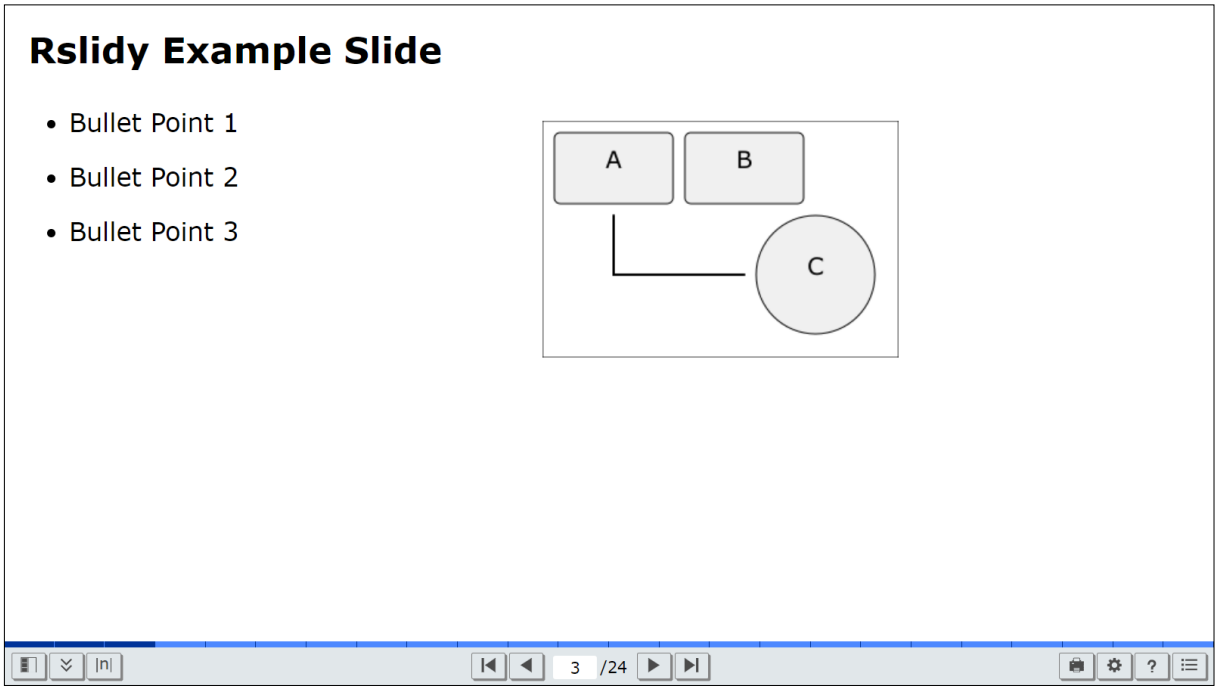


**Figure 5.1:** Rslidy: Example slide. The corresponding source code is shown in Listing 5.1.

```
<section>
  <h1>Rslidy Example Slide</h1>
  <div class="columns">
    <ul class="item-auto">
      <li>Bullet Point 1</li>
      <li>Bullet Point 2</li>
      <li>Bullet Point 3</li>
    </ul>
    <figure>
      <div class="figure" style="width:100%;">
        <img src="diagrams/diagram.svg" alt="Example Image" />
      </div>
    </figure>
  </div>
</section>

<style>
  .columns {
    display: flex;
    justify-content: space-between;
    align-items: flex-start;
    gap: 1.25rem;
  }
  .item-auto {
    flex: 1;
    margin-top: 3rem
  }
  figure {
    flex: 1;
    margin: 0;
  }
</style>
```

**Listing 5.1:** Rslidy: Example slide source code. The resulting slide is shown in Figure 5.1.

# Chapter 6

# Concluding Remarks

This survey provides an overview of web-based slide deck tools of different types, comparing their similarities and differences.

Text-based slide deck tools, built on Markdown, provide a simple and efficient way to create presentations without needing technical expertise, allowing users to focus on content. For developers, a good choice for a text-based slide deck would be Slidev. It supports Markdown for easy slide creation and offers live previews for immediate feedback while editing. With export options to PDF, PNG, and PowerPoint, it provides flexibility for sharing and presenting. Its integration with the Vue.js framework, makes it powerful for those familiar with the framework. However, Slidev requires a Node environment and relies on the Vue.js ecosystem, which makes it challenging for non-developers.

JavaScript-based slide deck tools, using HTML, CSS, and JavaScript, offer highly interactive and customisable presentations with animations and browser-based accessibility. Reveal.js stands out among JavaScript-based presentation frameworks due to its flexibility, robust features, and active community. It offers an extensive plugin ecosystem for customisation, an auto-animate feature for seamless transitions, and vertical slide navigation. Its also features an export function to PDF. Additionally, its active community and thorough documentation make it accessible even for users with moderate coding knowledge, ensuring ongoing support and frequent updates. However, as with other javaScript-based slide deck tools, some HTML, CSS, and JavaScript skills are required to fully use Reveal.js.

Hosted slide deck tools are accessible online without local installation, support real-time collaboration among users, have drag-and-drop editors, and integrate with productivity suites. Google Slides is an easily accessible, cloud-based slide deck tool perfect for collaboration. It allows multiple users to edit in real time, making it ideal for team projects. Features like version history enable users to track and revert changes, while offline editing ensures flexibility without internet access. Its simplicity and integration with Google Workspace make it a practical choice for quick and seamless presentations. However, limitations include the lack of SVG support, limited bullet formatting, and limited page numbering, which might be a drawback for users with advanced design needs.

Responsive slide decks are web-based slide decks which can adapt themselves to various display devices, by responding to different screen widths, and supporting mouse, touch, and keyboard input. Rslidy is the only slide deck tool designed explicitly for modern responsive layouts and interactivity. It features a responsive framework for the presentation, interactive navigation, an advanced image viewer, and live code integration. These capabilities make it ideal for web-based presentations with the focus on responsiveness and interaction. Rslidy requires knowledge of HTML, CSS, and JavaScript, since slide authors are responsible for ensuring that their slide content is responsive.

As technology continues to evolve, web-based slide decks are expected to play a larger role in improving the quality, interactivity, and reach of presentations. Their increasing integration with other online tools could also change how presentations are created and experienced in the future.

# Bibliography


AMS [2022]. *MathJax*. American Mathematical Society, 08 Jun 2022. https://mathjax.org/ (cited on pages 6, 10).

Apple [2003]. *Apple Keynote*. 07 Jan 2003. https://apple.com/in/keynote/ (cited on page 1).

Bang, Ole Petter [2023]. *Remark*. 27 Jun 2023. https://github.com/gnab/remark (cited on pages 5, 7).

Bauer, Gerald [2011]. *Slide Show(S9)*. 2011. https://slideshow-s9.github.io/ (cited on page 5).

Bublitz, Blaine and Eric Schoffstall [2024]. *Gulp*. 29 Mar 2024. https://github.com/gulpjs/gulp (cited on page 20).

Cone, Matt [2024]. *Markdown Guide*. 01 Oct 2024. https://markdownguide.org/ (cited on page 5).

Dalgleish, Mark [2020]. *Bespoke.js*. 08 Sep 2020. https://github.com/bespokejs/bespoke (cited on page 20).

El Hattab, Hakim [2024]. *reveal.js*. 15 May 2024. https://github.com/hakimel/reveal.js (cited on page 18).

El Hattab, Hakim and Owen Bossola [2018]. *slides*. 03 Oct 2018. https://slides.com/ (cited on page 18).

Fu, Anthony [2024]. *Slide Decks in HTML*. 11 Jun 2024. https://github.com/slidevjs/slidev (cited on page 10).

Gaskins, Robert, Tom Rudkin, and Dennis Austin [2024]. *Microsoft Powerpoint*. 2024. https://robertgaskins.com/powerpoint-history/sweating-bullets/gaskins-sweating-bullets-webpdf-isbn-9780985142414.pdf (cited on page 1).

Google [2024]. *Google Slides*. 2024. https://workspace.google.com/products/slides/ (cited on page 27).

Gruber, John [2004]. *Markdown*. 17 Dec 2004. https://daringfireball.net/projects/markdown/ (cited on page 5).

Halácsy, Péter, Peter Arvai, and Adam Somlai-Fischer [2009]. *Prezi*. 2009. https://prezi.com/ (cited on pages 19, 30).

Hattori, Yuki [2024]. *Marp*. 21 Apr 2024. https://github.com/marp-team/marp (cited on page 6).

Hipp, Patrick [2019]. *Slide Decks in HTML*. Survey Paper. Graz University of Technology, 25 Jan 2019. https://ftp.isds.tugraz.at/pub/papers/hipp-2019-01-25-survey-slide-decks.pdf (cited on pages 1, 5).

Hipp, Patrick [2021]. *Rslidy: Lightweight, Accessible, and Responsive HTML5 Slide Decks*. Master's Thesis. Graz University of Technology, 11 Sep 2021. https://ftp.isds.tugraz.at/pub/theses/phipp-2021-msc.pdf (cited on page 37).

Hiroto, Yuta [2021]. *Fusuma*. 01 Dec 2021. https://github.com/hiroppy/fusuma (cited on page 8).

Khan [2024]. *KaTeX*. Khan Academy, 02 Jul 2024. https://github.com/KaTeX/KaTeX (cited on page 6).

Ludvigsen, Holger and Espen H. Halvorsen [2011]. *Slidifier*. 2011. https://github.com/holgerl/Slidifier (cited on page 5).

MacFarlane, John, Martin Woodward, and Jeff Atwood [2024]. *CommonMark*. 28 Jan 2024. https://github.com/commonmark/commonmark-spec (cited on page 6).

Makeev, Vadim [2024]. *Shower*. 25 Oct 2024. https://github.com/shower/shower (cited on page 15).

Marp [2023]. *Marp for VS Code*. 28 Oct 2023. https://marketplace.visualstudio.com/items?itemName=marp-team.marp-vscode (cited on page 6).

MDX [2018]. *MDX*. Compositor and Vercel, 18 Jan 2018. https://github.com/mdx-js/mdx (cited on page 8).

Meta [2022]. *JSX*. 04 Aug 2022. https://facebook.github.io/jsx/ (cited on page 8).

Microsoft [2024]. *Microsoft Sway*. 2024. https://sway.cloud.microsoft/ (cited on page 33).

Nakajima, Pat and Dan Croak [2012]. *Slidedown*. 2012. https://github.com/nakajima/slidedown (cited on page 5).

Netlify [2024]. *Netlify*. 19 Oct 2024. https://netlify.com/ (cited on page 15).

Node [2024]. *Node.js*. 20 Nov 2024. https://nodejs.org/ (cited on pages 5, 8).

NPM [2024a]. *npm*. 03 Oct 2024. https://npmjs.com/ (cited on page 15).

NPM [2024b]. *npm bespoke plugins*. 20 Jul 2024. https://npmjs.com/search?q=keywords:bespoke-plugin (cited on page 20).

Raggett, Dave [2005]. *HTML Slidy: Slide Shows in XHTML*. Mar 2005. https://w3.org/2005/03/slideshow.html (cited on pages 1–2).

Raggett, Dave [2006]. *HTML Slidy: Slide Shows in XHTML*. 2006. https://w3.org/Talks/Tools/Slidy2 (cited on pages 1–2, 37).

Rslidy [2024]. *Rslidy: Lightweight, Accessible, and Responsive HTML5 Slide Decks*. 30 Jan 2024. https://github.com/tugraz-isds/rslidy (cited on page 37).

StackBlitz [2024]. *StackBlitz*. 10 Jan 2024. https://github.com/stackblitz (cited on page 10).

Szopka, Bartek and Henrik Ingo [2024]. *impress.js*. 26 Apr 2024. https://github.com/impress/impress.js (cited on page 19).

Taei, Payman [2024a]. *Visme*. 2024. https://visme.co/ (cited on page 30).

Taei, Payman [2024b]. *Visme*. 2024. https://www.visme.co/de/pricing/ (cited on page 30).

Troughton, Caleb [2016]. *Deck.js*. 04 May 2016. https://github.com/imakewebthings/deck.js (cited on page 17).

Verou, Lea [2022]. *PrismJS*. 23 Aug 2022. https://github.com/PrismJS/prism (cited on page 24).

Verou, Lea [2024]. *Inspire.js*. 17 Feb 2024. https://github.com/LeaVerou/inspire.js (cited on page 22).

Warström, Johnny, Niklas Ingvar, Kristoffer Renholm, and Thomas Toti [2012]. *Mentimeter*. 2012. https://mentimeter.com/ (cited on page 32).

You, Evan [2024]. *Vue.js*. 29 Aug 2024. https://github.com/vuejs/core (cited on page 10).

Zoho [2024]. *Zoho Show*. 2024. https://zoho.com/show/ (cited on page 28).