D.Y.PATIL COLLEGE OF ENGINEERING &TECHNOLOGY,

KASABA BAWADA, KOLHAPUR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

(Academic Year: 2024-25)

CASE STUDY

On

"Stripe Frontend Technology"

Submitted By:

Name	**Roll No**
Adarsh Ananda Nikam	16

Under the Guidance of:

Guid Name: Prof. Ranjita S. Jadhav

Class: TY (CSE) Div.: "C" Batch: T1

Introduction :

Stripe is a well-known online payment platform that helps businesses accept payments safely and easily. Its website is designed to be modern, fast, and user-friendly, making it a great example for studying web technologies.

The frontend of Stripe is built using HTML, CSS, and JavaScript, which are the core technologies for any website. It also uses advanced tools like React.js and Next.js to make the website more interactive and efficient. CSS frameworks like Tailwind CSS help in styling the site, making it look attractive and work smoothly on all devices.

Stripe's website also includes animations and transitions using tools like GSAP and WebGL, which make the user experience more engaging. It is designed to load quickly by using smart techniques like lazy loading and code splitting. The website is also responsive, meaning it works well on mobile phones, tablets, and computers. With a clean design and smooth performance, Stripe sets a high standard for modern websites.

Objectives of the Case Study :

1. Analyze the frontend technologies and frameworks used in Stripe.
2. Understand how UI/UX principles improve user experience.
3. Explore the techniques used to optimize performance.
4. Study the interactive animations and smooth transitions in Stripe’s UI.
5. Identify the best frontend practices that developers can learn from.

Frontend Technologies Used :

Stripe leverages the latest frontend technologies to enhance performance, interactivity, and usability. Some of the key technologies include:

1. HTML5 & CSS3

- Provides the structural foundation of the website.
- Uses Tailwind CSS for utility-first styling.
- Implements CSS Flexbox and Grid for responsive layouts.

2. JavaScript & Frameworks

- React.js → Builds interactive and reusable UI components.
- Next.js → Enables server-side rendering (SSR) for faster page loads.
- TypeScript → Ensures better code maintainability and security.

3. Animations & Graphics

- GSAP (GreenSock Animation Platform) → Provides smooth and professional animations.
- Three.js & WebGL → Enables 3D effects and visual elements.
- SVG animations → Improves performance with vector graphics.

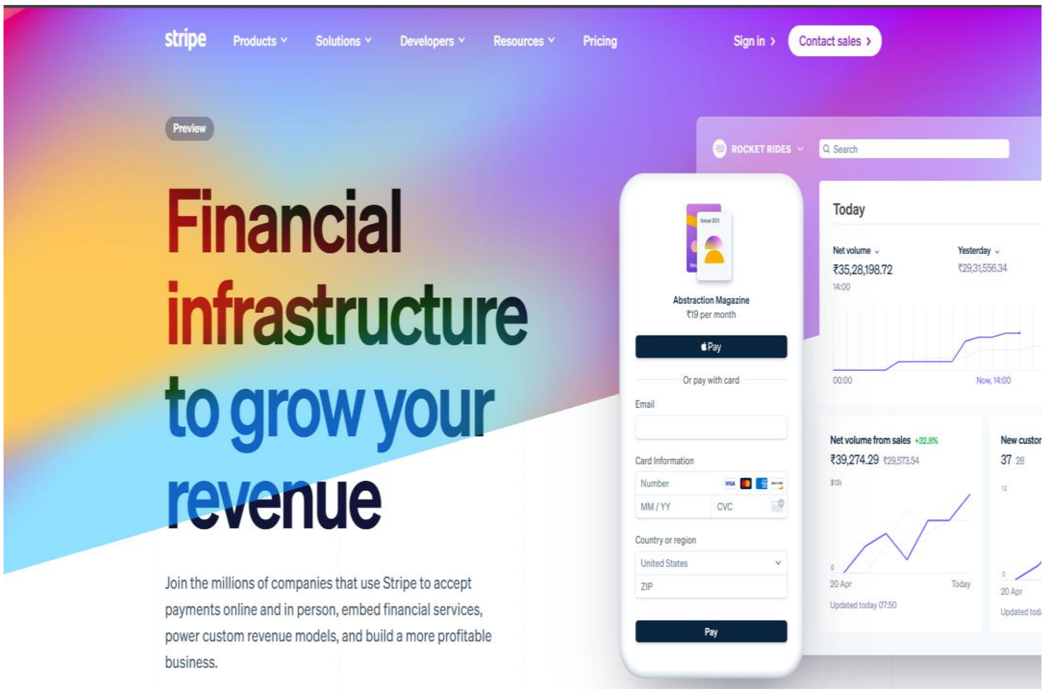
4. Performance Optimization

- Lazy Loading → Loads images and scripts only when needed.
- Code Splitting → Reduces load time by splitting JavaScript files.
- Caching & Service Workers → Enhances offline capabilities and speeds up page reloads.

5. API & Data Handling

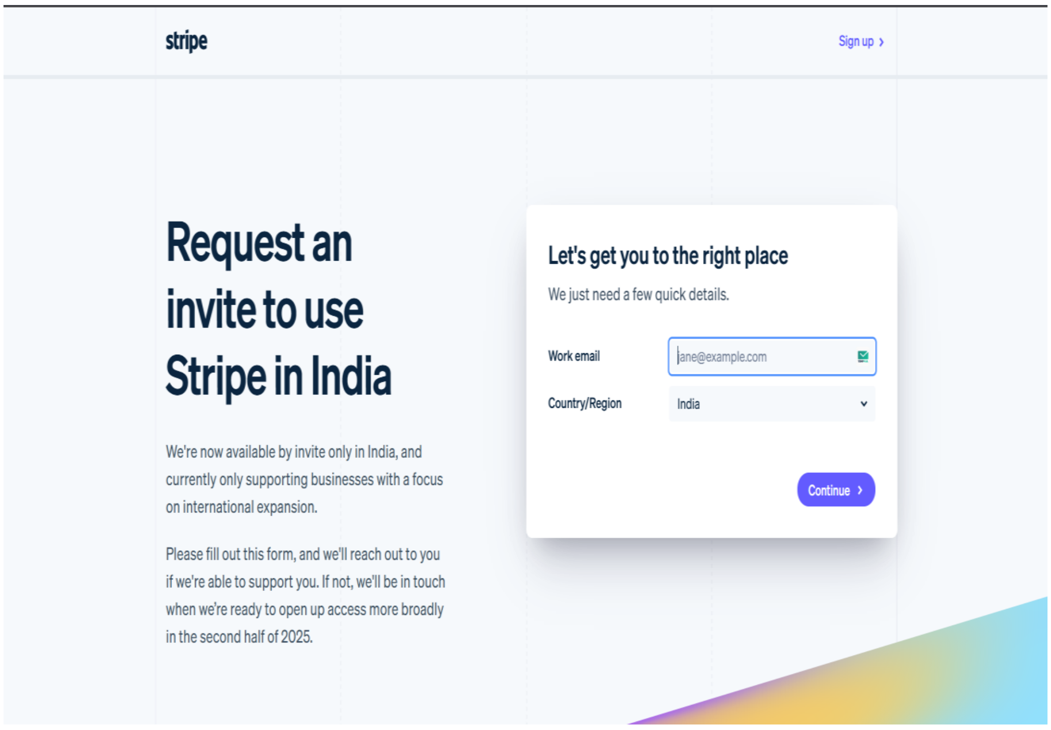
- GraphQL → Fetches optimized data to improve API efficiency.
- WebSockets → Enables real-time updates and interactions.

Snapshots :



Financial infrastructure to grow your revenue

Join the millions of companies that use Stripe to accept payments online and in person, embed financial services, power custom revenue models, and build a more profitable business.



Request an invite to use Stripe in India

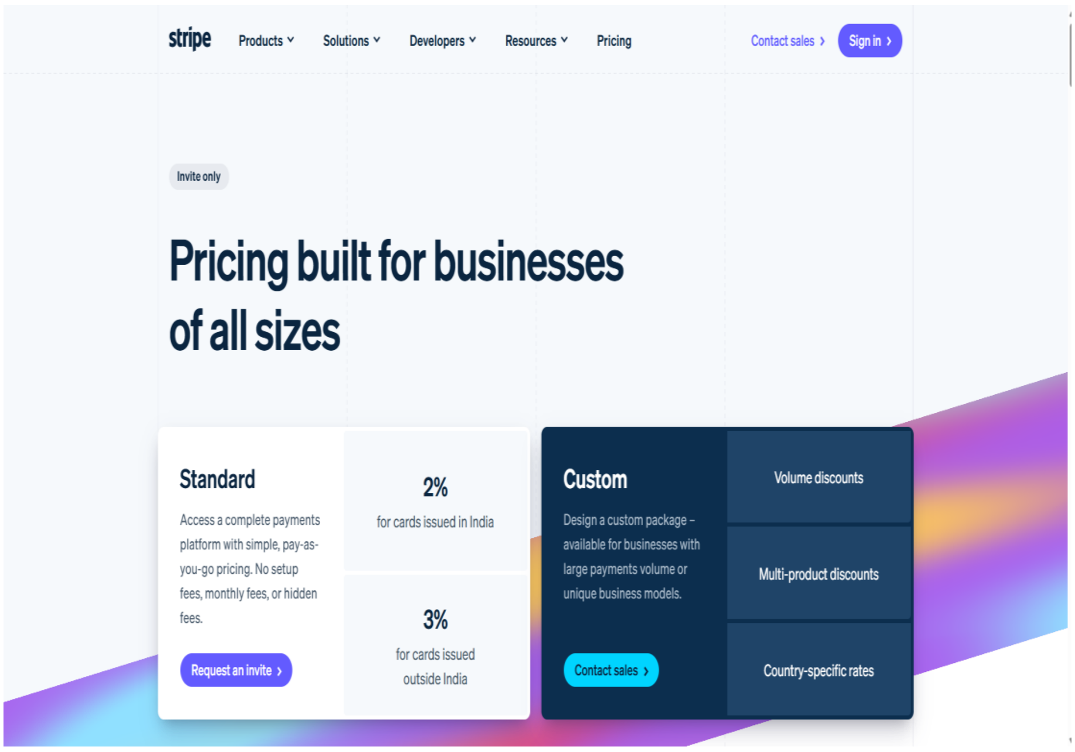
We're now available by invite only in India, and currently only supporting businesses with a focus on international expansion.

Please fill out this form, and we'll reach out to you if we're able to support you. If not, we'll be in touch when we're ready to open up access more broadly in the second half of 2025.

Let's get you to the right place

We just need a few quick details.

Work email	jane@example.com
Country/Region	India



Pricing built for businesses of all sizes

Standard

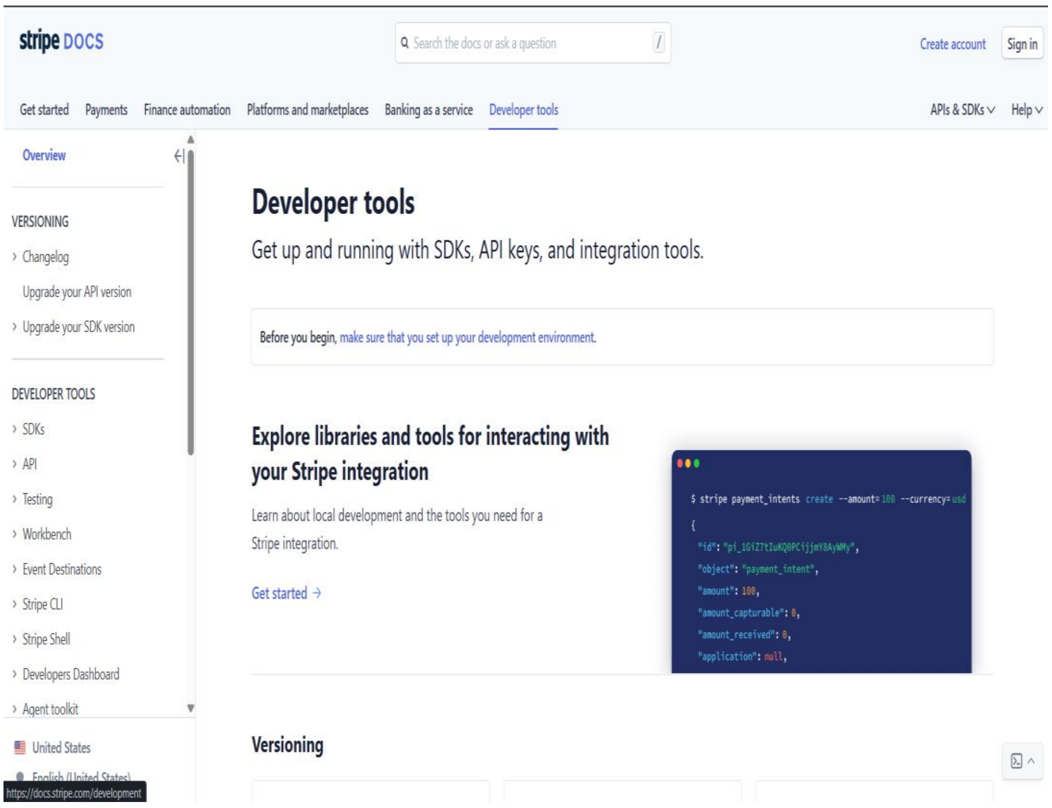
Access a complete payments platform with simple, pay-as-you-go pricing. No setup fees, monthly fees, or hidden fees.

2% for cards issued in India

3% for cards issued outside India

Custom

Design a custom package – available for businesses with large payments volume or unique business models.



Developer tools

Get up and running with SDKs, API keys, and integration tools.

Before you begin, make sure that you set up your development environment.

Explore libraries and tools for interacting with your Stripe integration

Learn about local development and the tools you need for a Stripe integration.

```
$ stripe payment_intents create --amount=100 --currency=usd
{
  "id": "pi_1GiZ7tIuKQ0PCijjmY8AyWMy",
  "object": "payment_intent",
  "amount": 100,
  "amount_capturable": 0,
  "amount_received": 0,
  "application": null,
```

Versioning

Source Code Study :

HTML Structure

- Uses semantic HTML tags like <header>, <section>, and <footer> for better organization and accessibility.
- Well-structured layout makes the code easy to maintain.
- Uses lazy loading to load images only when needed, making the site faster.
- Includes ARIA attributes to help visually impaired users navigate easily.

CSS & Styling

- Uses Tailwind CSS for a clean and responsive design.
- Flexbox and CSS Grid help arrange content neatly.
- CSS animations and transitions make interactions smooth.
- Supports dark mode for a better viewing experience.
- Media queries ensure the site works well on all screen sizes.

JavaScript & React Components

- React.js and Next.js make the website interactive and fast.
- Event listeners enable smooth animations and effects (like button clicks and scrolling).
- Next.js server-side rendering (SSR) helps pages load faster.
- WebSockets allow real-time updates for payments.
- Lazy loading and code splitting improve speed by loading only necessary parts of the site.
- GraphQL APIs fetch data quickly and efficiently.

Advanced Frontend Technologies :

- GSAP (GreenSock) – Creates high-performance animations.
- Three.js & WebGL – Enables 3D graphics and interactive elements.
- Optimized SVG Animations – Provides smooth vector animations.
- Progressive Web App (PWA) Features – Enhances mobile experience.
- Dark Mode Support – Provides a user-friendly theme toggle.
- Micro-interactions – Small but impactful UI animations.
- WebAssembly – Reduces JavaScript execution time, making interactions faster.

What's Unique About Stripe's Frontend ?

Stripe's frontend stands out due to its smooth and seamless animations, which are powered by GSAP. These animations create an engaging and interactive user experience. The website is highly responsive, thanks to Next.js and Tailwind CSS, ensuring it adapts well to different screen sizes and devices.

Performance is a key focus, with server-side rendering (SSR) making the website load faster and more efficiently. Stripe also incorporates interactive 3D effects using Three.js and WebGL, which enhance visual appeal and user engagement.

Real-time updates are another highlight, achieved using WebSockets, which enable instant data updates without needing page refreshes. Additionally, AI-powered recommendations personalize the user journey, improving navigation and overall experience on the platform.

Conclusion :

Stripe’s frontend architecture is a benchmark in modern web development. It uses HTML for structuring web pages with semantic elements like <header>, <section>, and <footer>, ensuring accessibility and easy maintenance. CSS plays a key role in styling, with Tailwind CSS providing a flexible and responsive design while animations and transitions enhance user interaction.

For advanced functionalities, JavaScript, React.js, and Next.js are used to create a dynamic and smooth experience. GSAP enables high-performance animations, while Three.js and WebGL bring interactive 3D effects. Server-side rendering (SSR) in Next.js ensures fast performance, and WebSockets allow real-time updates. By combining these technologies, Stripe delivers a seamless, visually appealing, and high-performance web experience.

References :

- Stripe Official Website: https://stripe.com
- Frontend Documentation & Blogs