



Zion: Revolutionizing Text IP Management within Story Ecosystem

Zion is a revolutionary platform that leverages Story Protocol to transform the way personal text-based intellectual property is managed and monetized.

Problem Definition: Challenges in Text IP Management



Personal data protection

Text-based IP often contains sensitive personal information that individuals are hesitant to share publicly, limiting the ability to leverage and monetize their content.



Lack of monetization opportunities

Creators of high-quality text-based content, such as resumes and portfolios, have limited options to monetize their work, missing out on potential revenue streams.



Inadequate IP protection

Existing platforms do not provide robust mechanisms to protect the intellectual property rights of text-based content, leading to unauthorized use and lack of attribution.

The combination of personal data protection concerns, limited monetization avenues, and inadequate IP safeguards presents significant challenges for individuals and organizations seeking to leverage the value of their text-based intellectual property.

Zion's Solution Overview



Zion Platform Overview

Zion's platform integrates seamlessly with Story Protocol to offer a comprehensive solution for managing and monetizing text-based intellectual property (text IP). The platform enables users to securely upload their personal documents, such as resumes, cover letters, and portfolios, and have them transformed into valuable, protectable digital assets.

Zion's Solution Overview

Secure Storage and Tokenization

When a user uploads their text IP to the Zion platform, the content is stored on IPFS (InterPlanetary File System) for secure and decentralized storage. The platform then generates a unique token representing the text IP, which is registered on the Story Protocol blockchain. This tokenization process ensures the content's provenance, authenticity, and ownership are transparently recorded on the blockchain.

AI-Powered Content Optimization

Zion's platform leverages advanced AI algorithms to analyze the uploaded text IP and provide users with optimized versions of their content. The AI agent identifies areas for improvement, such as grammar, formatting, and content structure, and generates enhanced versions of the documents. Users can then choose to adopt the AI-generated improvements or keep their original content.

Monetization and Royalty Tracking

The Zion platform enables users to monetize their text IP by making it available for reference or licensing to other platform users. When a user's content is referenced or licensed, the platform automatically tracks the usage and distributes royalties to the original content creator through the Story Protocol's royalty management modules. This ensures fair compensation for the creator's intellectual property.

Transparent Rights Management

All references, licenses, and royalty distributions related to the text IP are recorded on the Story Protocol blockchain. This provides a transparent and immutable record of the usage and ownership of the content, protecting the creators' rights and enabling seamless management of the text IP ecosystem.

The Value Proposition: Why Zion?

Protect Personal Data

Zion safeguards personal information in text-based intellectual property, such as résumés, by leveraging blockchain technology to control data access and visibility.

Monetization Opportunities

Zion enables content creators to monetize their text-based IP, such as well-written résumés, by allowing others to access and use them in exchange for royalty payments.

Strengthen Employer-Candidate Connections

Zion connects employers with a pool of high-quality candidates by providing access to a marketplace of verified, reusable text-based IP, benefiting both parties.

Transparent IP Management

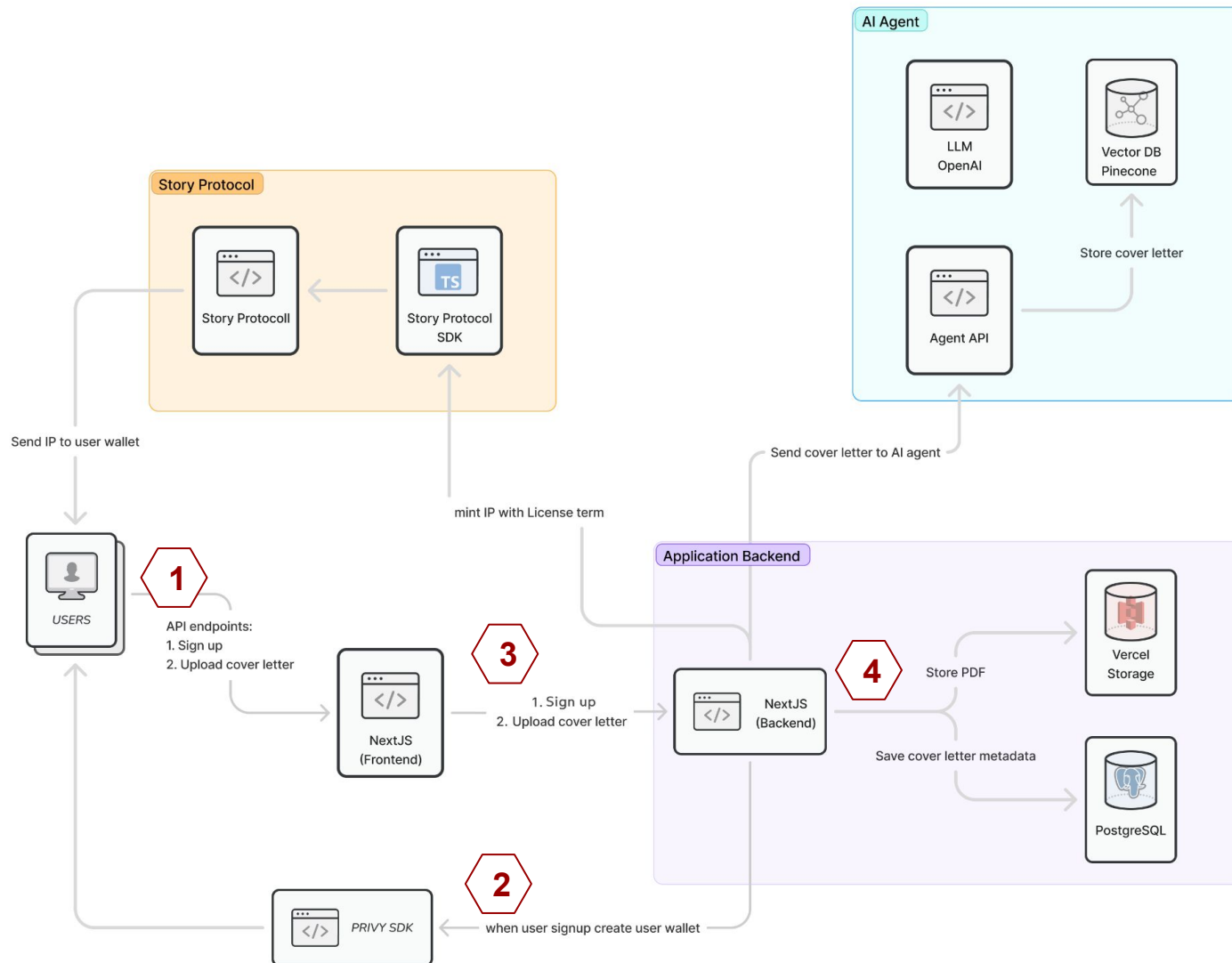
Zion's integration with Story Protocol ensures transparent ownership, licensing, and royalty tracking for text-based IP, empowering creators and users alike.

Why Blockchain?

Blockchain technology provides the immutability, transparency, and decentralized trust that are essential for managing and monetizing personal text-based intellectual property.

Technical Architecture

Upload cover letter



Upload Cover Letter

1. User Interaction Begins

- A user signs up and logs into the platform through the **Next.js (Frontend)** interface.
- This interface provides access to API endpoints:

1. **Sign up**
2. **Upload cover letter**

2. Wallet Creation via Privy SDK

- Upon signup, the frontend uses **Privy SDK** to automatically create a **user wallet**.
- This wallet will later receive minted IP assets.

3. Uploading the Cover Letter

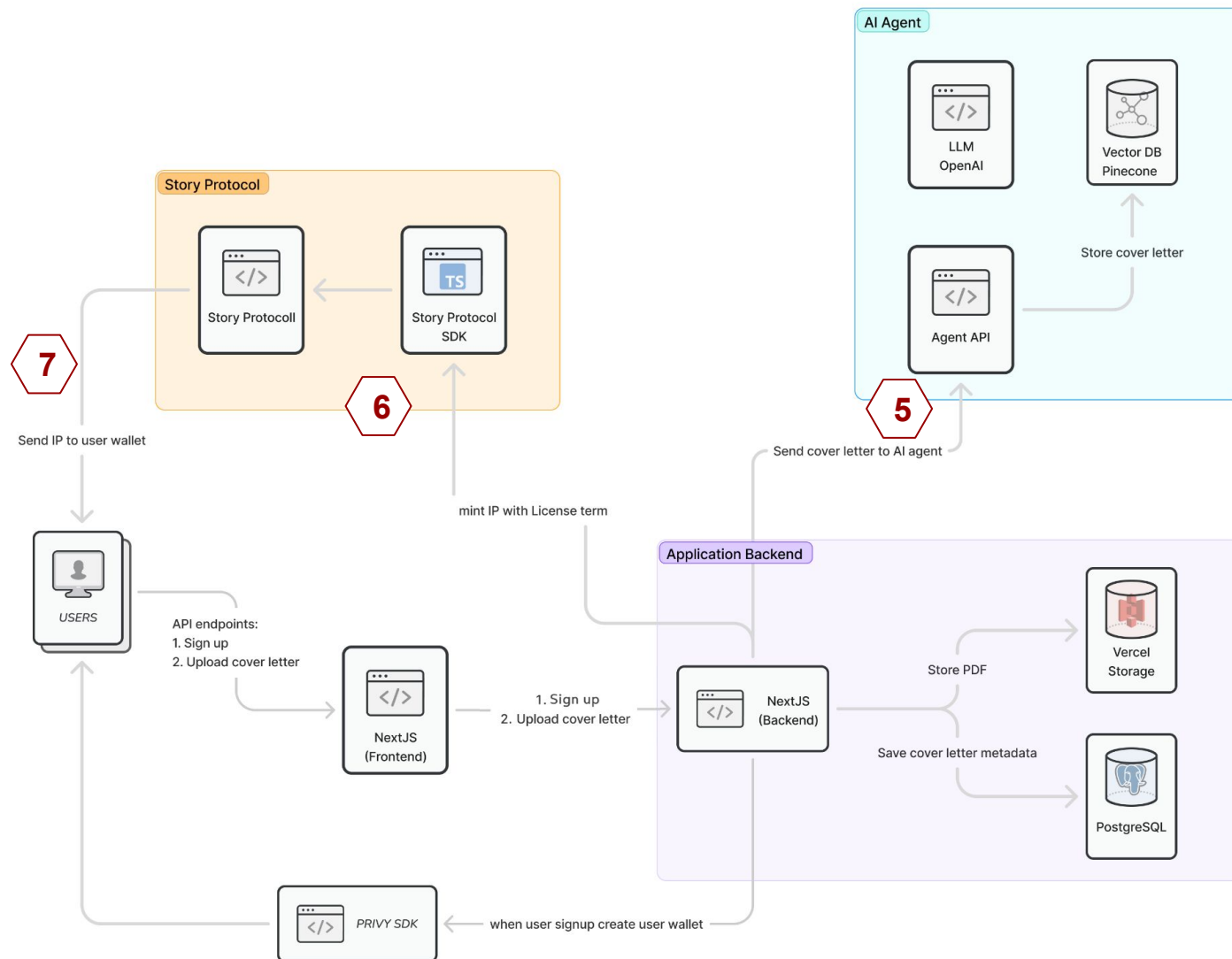
- The user uploads a cover letter through the frontend.
- The cover letter is sent to the **Next.js Backend** (Application Backend).

4. Backend Storage & Metadata Handling

- The **cover letter PDF** is:
 - Stored in **Vercel Storage**
 - Its **metadata** (e.g., Job description, tenure etc) is saved in **PostgreSQL**

Technical Architecture

Upload cover letter



5. AI Agent Integration

- The backend also sends the uploaded cover letter to the **AI Agent**:
 - Via the **Agent API**
 - The cover letter is:
 - Stored in a **Vector DB (Pinecone)** for future similarity search and training
 - Can be optionally used for fine-tuning or prompt engineering with **OpenAI LLM**

6. Story Protocol Registration

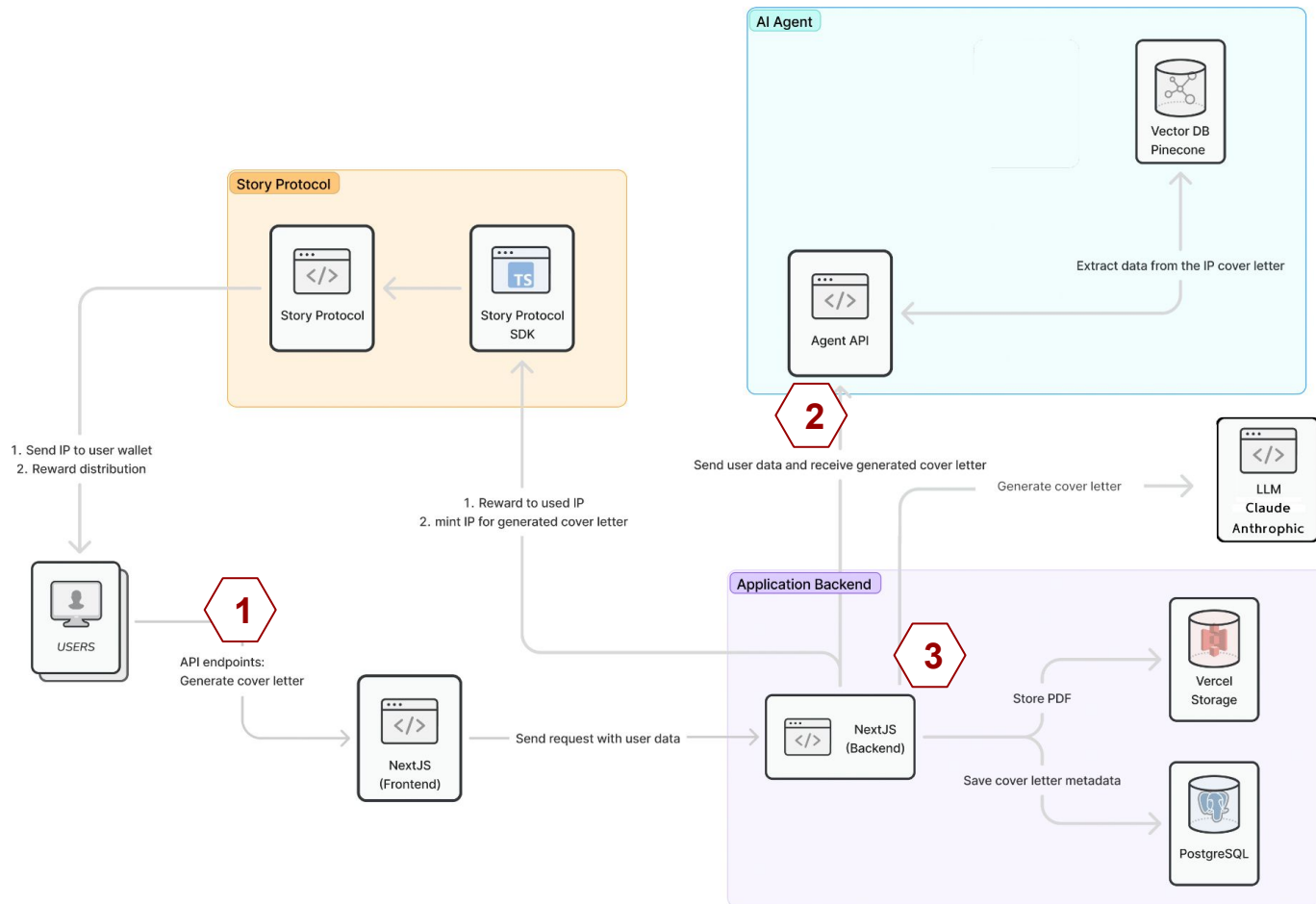
- The backend uses the **Story Protocol SDK** to:
 - Mint the uploaded cover letter as **IP** (intellectual property), key value of pdf link stored simultaneously
 - Assign **License Terms** (e.g., usage rights, visibility scope)

7. IP Minting & Distribution

- The **Story Protocol SDK** communicates with the **Story Protocol** to register the IP on-chain.
- Once minted, the **IP asset** is:
 - Sent to the **user's wallet** (created via Privy SDK)

Technical Architecture

Generate cover letter



Generate Cover Letter

1. User Initiates Cover Letter Generation

- The user clicks “**Generate Cover Letter**” in the **Next.js frontend**.
- A request is made to the **Application Backend** containing:
 - User profile data
 - Job description or target position

2. Backend Sends Request to AI Agent

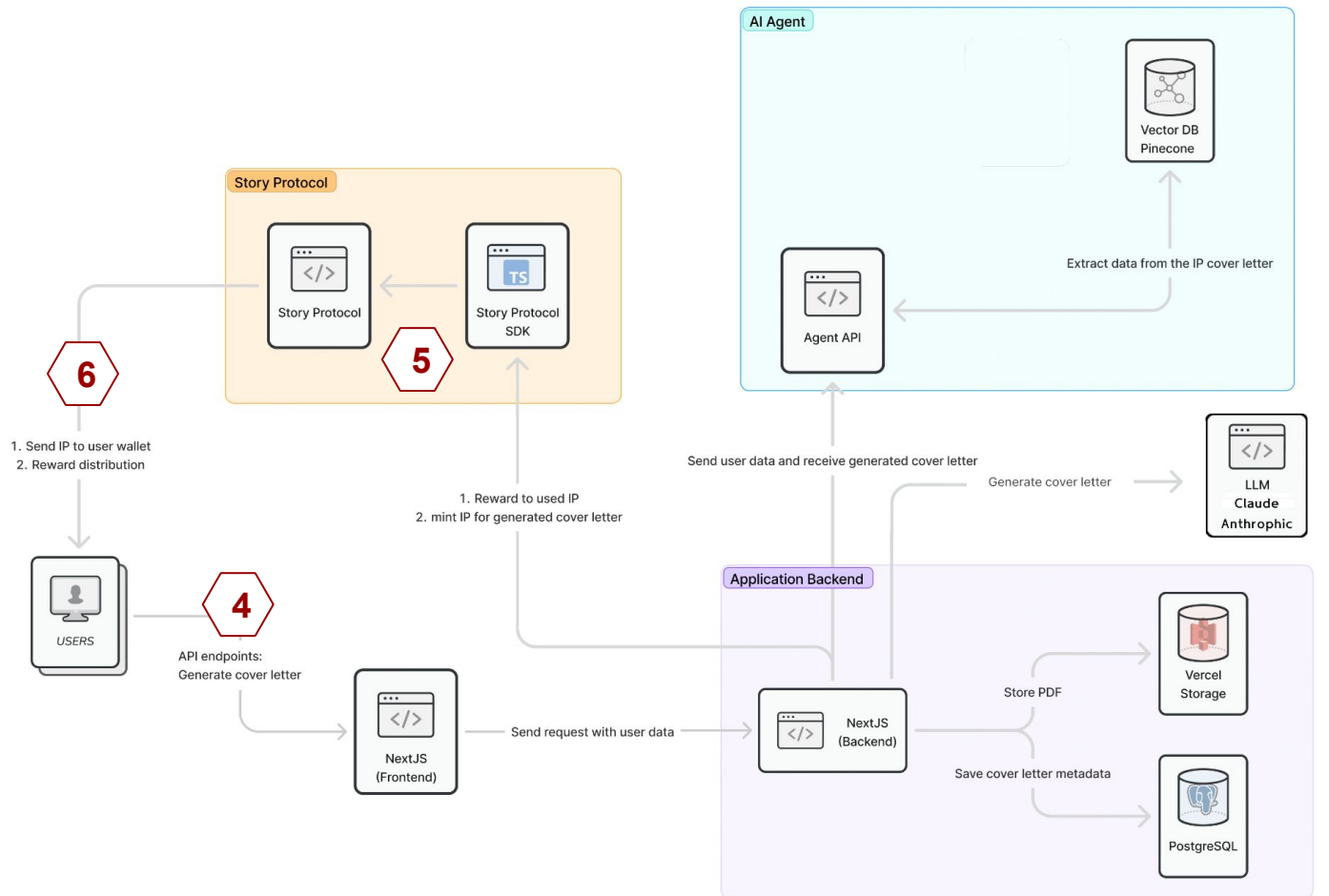
- The backend forwards this user data to the **Agent API**.
- The **Agent API** uses:
 - **Pinecone Vector DB** to retrieve and extract relevant data from previously stored IP resumes.
 - **OpenAI LLM** to generate a personalized cover letter based on:
 - Extracted ip cover letter information
 - The user’s target job or role

3. Generated Cover Letter Returned to Backend

- The generated cover letter is returned to the **Next.js backend**.
- The backend:
 - **Stores the PDF** in **Vercel Storage**
 - **Saves metadata** (author, IP references, timestamps, etc.) in **PostgreSQL**

Technical Architecture

Generate cover letter



Generate Cover Letter

4. Minting the Cover Letter as an IP

- The backend uses the **Story Protocol SDK** to:
 - Mint the newly generated cover letter as a new IP asset**
 - Distribute rewards** to any **previous IPs** (cover letter) that were used as input in the AI generation process

5. Story Protocol Handles On-Chain IP Registration

- The **Story Protocol (OpenAI)** receives instructions from the SDK to:
 - Register the new cover letter IP
 - Process attribution and **reward the creators** of reused IPs

6. Send IP to User Wallet

- After minting, the new **IP asset** is sent to the user's wallet (created during sign-up).
- Users retain ownership and can manage or license the IP asset.

Story Protocol Integration

IP Asset

Zion uses the IP Asset of Story Protocol to register each text-based IP (e.g., résumé) as a unique asset on the blockchain. This enables transparent ownership and tracking of the content.

```
await client.ipAsset.mintAndRegisterIpAssetWithPillTerms({
  spgNftContract:
'0xc32A8a0FF3beDDDa58393d022aF433e78739FAbc',
  licenseTermsData: [{ terms:
creativeCommonsAttribution }],
  ipMetadata: {
    ipMetadataURI: ipUri,
    ipMetadataHash: ipMetaHash,
    nftMetadataHash: nftMetaHash,
    nftMetadataURI: nftUri,
  },
  txOptions: { waitForTransaction: true },
})
```

5

License Module

The License module allows Zion to define and manage the licensing terms and conditions for the use of the registered text IPs. This ensures appropriate attribution and royalty distribution for content creators.

```
await
client.ipAsset.mintAndRegisterIpAndMakeDerivativeWithL
icenseTokens({
  spgNftContract:
'0xc32A8a0FF3beDDDa58393d022aF433e78739FAbc',
  licenseTokenIds: licenseTokenIds,
  ipMetadata: {
    ipMetadataURI: ipMetaUri,
    ipMetadataHash: ipMetaHash,
    nftMetadataHash: nftMetaHash,
    nftMetadataURI: nftMetaUri,
  },
  maxRts: 100_000_000,
  txOptions: { waitForTransaction: true },
  recipient: userWallet,
})
```


Royalty Module

Zion integrates the Royalty module of Story Protocol to automatically distribute royalties to the original creators whenever their content is referenced or utilized by other users. This provides a seamless monetization mechanism.

```
await client.royalty.payRoyaltyOnBehalf({
  receiverIpId: PARENT_IP_ID,
  payerIpId: ZION_ADDRESS,
  token: WIP_TOKEN_ADDRESS,
  amount: parseEther('2'),
  txOptions: { waitForTransaction: true },
})

await client.royalty.claimAllRevenue({
  ancestorIpId: PARENT_IP_ID,
  claimer: PARENT_IP_ID,
  childIpIds: [],
  royaltyPolicies: [RoyaltyPolicyLRP],
  currencyTokens: [WIP_TOKEN_ADDRESS],
})
```

User Experience Flow - Sign up/Login



Create Account

Enter your information below to create your account

Name

Email


Password

Confirm Password

[Sign Up](#)

Sign up

just name, email, and password
→ wallet auto-generated & linked by **PRIVY SDK**



Login to your account

Enter your email below to login to your account

Email

Password

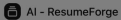
[Forgot your password?](#)

[Login](#)

Don't have an account? [Sign up](#)

Login

Login with registered email address and password



Login to your account

Enter your email below to login to your account

Email

Password

[Forgot your password?](#)

[Login](#)

Don't have an account? [Sign up](#)

Email Verification

Please enter the verification code sent to jeongahan@mrhi.or.kr

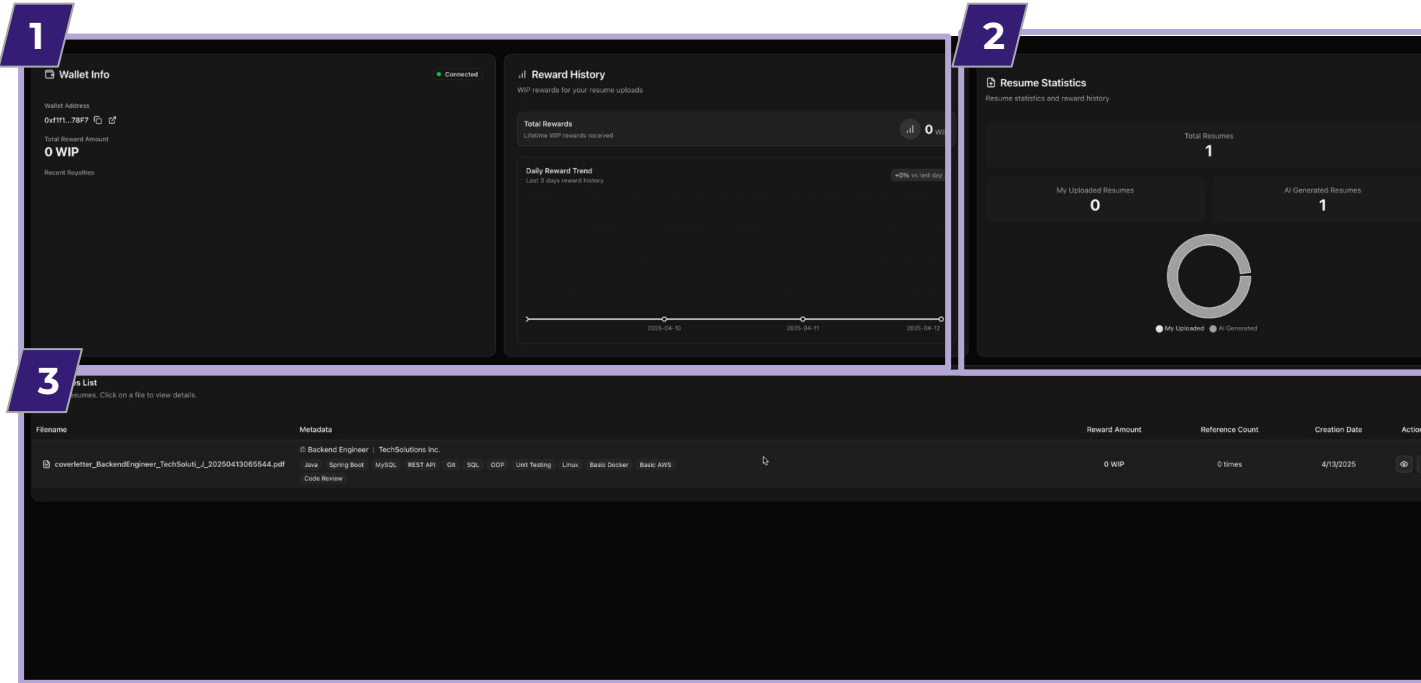
Verification Code

[Verify](#)

Verification

Email verification is required at sign-up/login to protect wallet-linked accounts. Auth.js, a session-based SDK, ensures secure authentication throughout.

User Experience Flow - Dashboard



Dashboard - each section

Section 1. Wallet info

- Wallet Address
- Total Reward Amount
- Transaction history
- initially empty

Section 2. Cover letter statistics

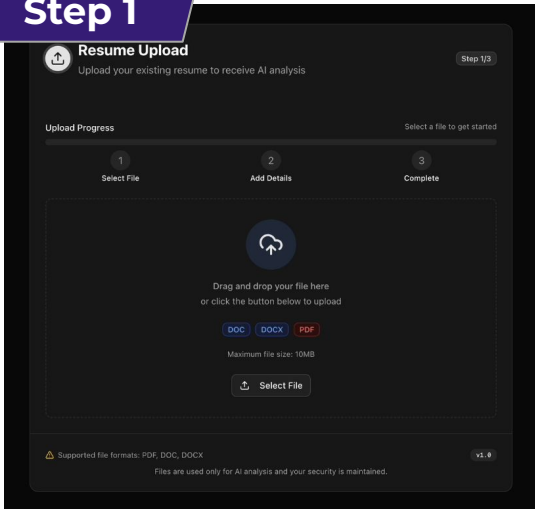
- Summary of Uploaded and AI-Generated Cover Letters

Section 3. Uploaded resume

- View all cover letters registered on-chain as intellectual property (IP)
- Reward, Reference Count, Created Date, View / Download options available

User Experience Flow - Upload & Reward

Step 1



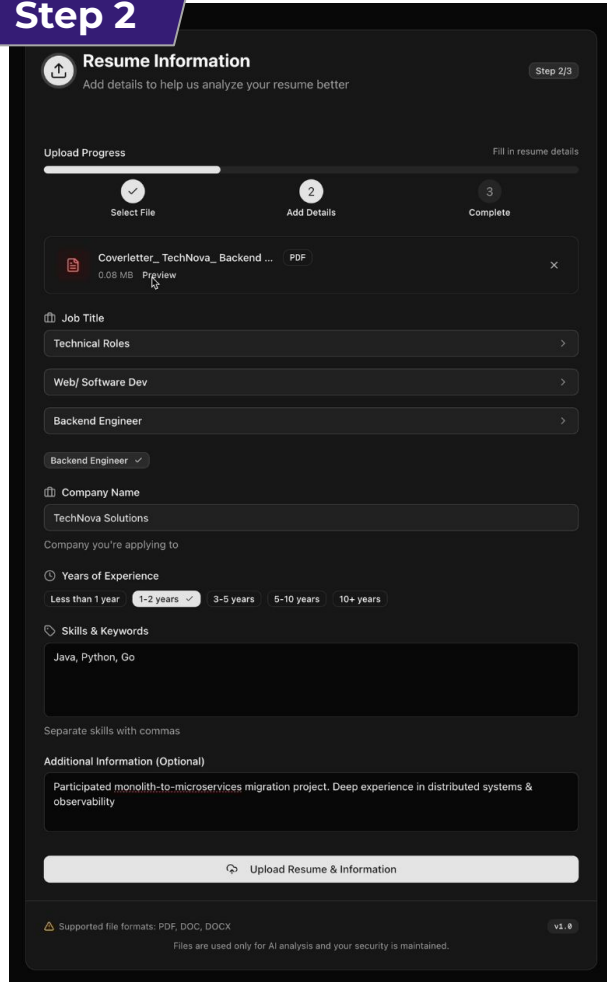
Resume Upload
Upload your existing resume to receive AI analysis

Upload Progress: 1 Select File, 2 Add Details, 3 Complete

Drag and drop your file here or click the button below to upload

Supported file formats: PDF, DOC, DOCX

Step 2



Resume Information
Add details to help us analyze your resume better

Upload Progress: 1 Select File, 2 Add Details, 3 Complete

Coverletter_TechNova_Backend ... PDF

Job Title: Technical Roles, Web/ Software Dev, Backend Engineer, Backend Engineer ✓

Company Name: TechNova Solutions

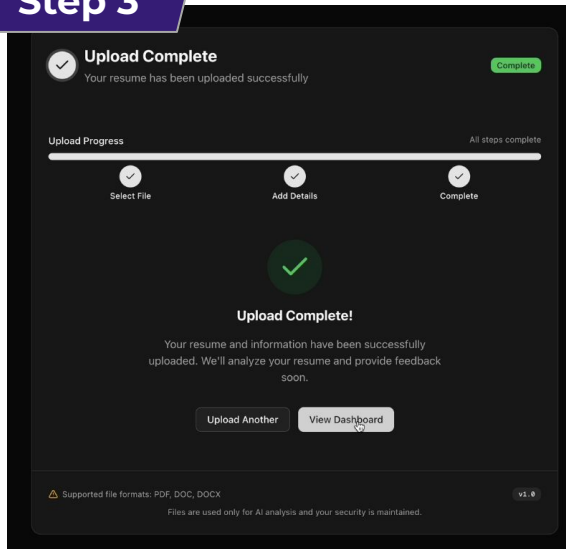
Years of Experience: Less than 1 year, 1-2 years, 3-5 years, 5-10 years, 10+ years

Skills & Keywords: Java, Python, Go

Additional Information (Optional): Participated monolith-to-microservices migration project. Deep experience in distributed systems & observability

Upload Resume & Information

Step 3



Upload Complete
Your resume has been uploaded successfully

Upload Progress: All steps complete

Upload Complete!

Your resume and information have been successfully uploaded. We'll analyze your resume and provide feedback soon.

Upload Another View Dashboard

Step 1. Select File

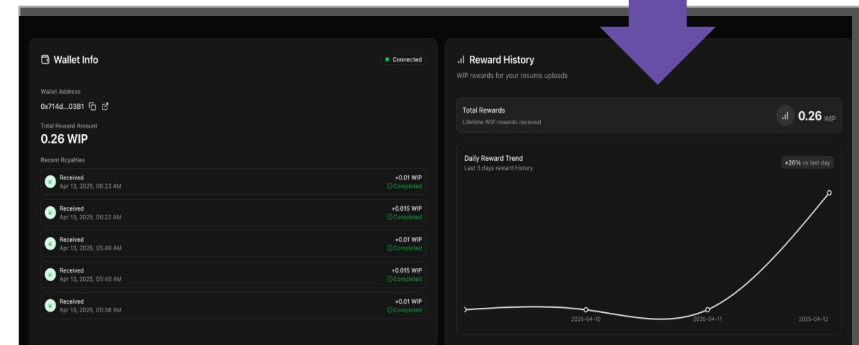
- Upload user generated cover letter

Step 2. ADD details

- Write job title, skills, and other info
- The uploaded cover letter is **classified into the Pinecone dataset** as metadata

Step 3. Complete

The **cover letter is now on-chain IP**, ready to be referenced by others and **earn Story Tokens through its contributions**



Reward System

Reward granted when each time your on-chain cover letter is referenced, based on contribution.

User Experience Flow - Generate & Reward

Step 1 & 2

Basic Information

Job Title

Technical Roles

Web/ Software Dev

Backend Engineer

Backend Engineer

Years of Experience

Less than 1 year

2-5 years

6-10 years

10+ years

Self Introduction

I am a junior backend engineer with a strong foundation in software development and a passion for building reliable and efficient systems. I have experience in developing web applications and APIs, with a focus on clean code and best practices. My background in computer science and hands-on experience with various backend technologies have given me a solid understanding of software development principles.

Motivation

I am excited about the opportunity to contribute to your development team and grow as a backend engineer. I believe my technical skills and enthusiasm for learning can add value to your projects. I am particularly interested in working on scalable systems and learning from experienced developers.

Relevant Experience

With one year of experience, I have worked on web application development. Key technologies: Java, Spring Boot, MySQL, REST API, Git. Key achievements: Developed 3 RESTful APIs, improved application performance by 20%, implemented automated testing, participated in code reviews.

Future Aspirations

I aim to develop my backend engineering expertise at your company. I am particularly interested in learning about system architecture, performance optimization, and best practices in software development.

Additional Information (Optional)

Skills & Keywords

Java, Spring Boot, MySQL, REST API, Git, SQL, OOP, Unit Testing, Linux, Basic Docker, B

Target Company

TechSolutions Inc.

Department

Backend Engineering

Custom Instructions

Please create a resume that emphasizes fundamental backend development skills and potential. Focus on: 1) Basic backend development experience, 2) API development knowledge, 3) Code quality and testing capabilities, 4) Learning potential in software development.

Introduction

I am a Computer Science graduate with a focus on backend development, applying for the Backend Engineer position at TechSolutions Inc. My journey in software engineering began during my studies when I discovered the elegant complexity of backend systems and their critical role in application performance. Throughout my early career, I have maintained a commitment to writing clean, maintainable code while focusing on system reliability. With one year of professional experience, I have developed a methodical approach to problem-solving and a keen eye for detail that helps me build robust backend solutions. My colleagues often describe me as analytical and collaborative, as I value knowledge sharing and constructive feedback during code reviews. I approach each project with both technical precision and an understanding of how backend systems support broader business requirements. My passion for backend development stems from seeing how well-designed APIs and efficient database interactions can dramatically improve application performance and user experience.

Motivation and Fit

I am writing to express my genuine interest in joining TechSolutions Inc., a company whose reputation for technical excellence and innovative backend solutions has caught my attention. After researching your company's projects and technology stack, I was particularly impressed by your approach to building scalable systems and your commitment to modern development practices. The challenges your backend team tackles—from designing efficient APIs to optimizing database performance—align perfectly with the areas I'm most passionate about developing further.

What draws me to TechSolutions Inc. is your

Model

claude-3-7-sonnet-20250219

Temperature

0.7

Maximum Length

4000

Top P

0.9

Processing

Stop Generation

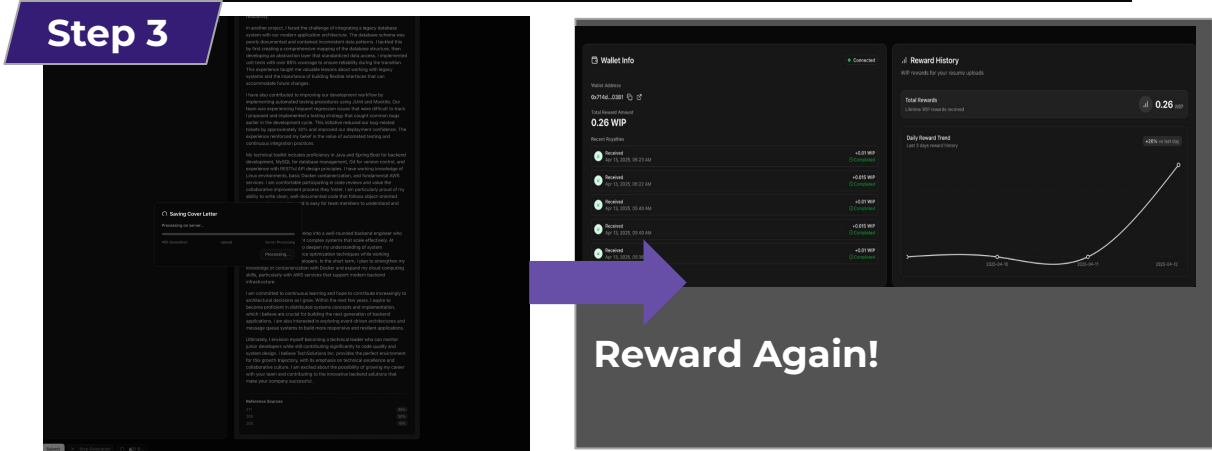
Generate cover letter

Step 1. Users fill out a structured form with their information, experience, motivation and future aspirations. Optional fields allow for fine-tuning based on target companies or specific keywords.

Step 2. AI generates a full-length, natural cover letter by referencing relevant data from previously registered cover letters in the database—both **user-submitted** and **AI-generated**.

Step 3. Users can review their generated cover letter for uploading on this site; when they click "Save," it is **registered on-chain as IP**, and the system tracks content attribution and rewards original contributors with Story tokens.

- All wallet-linked actions are securely managed via Auth.js and transparently recorded on-chain to ensure fair and verifiable reward distribution.



Conclusion and Next Steps



Key Takeaways

Summarized the core challenges in text IP management, Zion's innovative solution leveraging Story Protocol, and the key benefits for content creators, job seekers, and employers.



Future Expansion

Outlined Zion's roadmap for expanding supported content types and exploring new market opportunities, demonstrating the platform's scalability and long-term vision.



Collaboration Opportunities

Invited the audience to explore further possibilities for partnership, integration, or adoption of the Zion platform within their own ecosystems and workflows.

Zion's comprehensive approach to text IP management, powered by the Story Protocol blockchain, has the potential to transform how individuals and organizations leverage their valuable content. By addressing the core challenges and delivering a compelling value proposition, Zion is poised to revolutionize the way personal intellectual property is protected, shared, and monetized. We welcome the audience to join us in further exploring the opportunities presented by this innovative solution.