SRS Hamsaye

Hamsaye, a pre-launch self-storage startup, uses a peer-to-peer model to connect customers with individuals willing to rent out their unused space, bringing storage locations closer to consumers.

Contents

Introduction

1.1 Purpose

1. Provide a Convenient and Affordable Alternative to Traditional Self-Storage

- Hamsaye aims to revolutionize the storage industry by offering a more convenient and affordable alternative to traditional self-storage facilities. Their platform connects individuals with excess storage space in their homes or garages with those who need temporary or long-term storage solutions.

2. Empower Individuals to Monetize Their Unused Space

- Hamsaye provides a valuable platform for individuals to monetize their unused storage space, generating passive income from their underutilized assets. This concept of sharing underutilized resources aligns with the sharing economy movement, promoting a more sustainable and equitable use of resources.

3. Enhance Accessibility and Reduce Environmental Impact

- By connecting renters with storage options in their immediate vicinity, Hamsaye eliminates the need for long commutes to traditional storage facilities, saving time and reducing fuel consumption. This localized approach also contributes to a more sustainable storage model, minimizing environmental impact.

4. Foster Community Connections and Sense of Belonging

- Hamsaye facilitates interactions between renters and storage providers, fostering a sense of community and belonging within neighborhoods. This connection strengthens social ties and promotes a more neighborly atmosphere.

5. Provide Safe and Secure Storage Solutions

- Hamsaye prioritizes the safety and security of both renters and storage providers. The platform employs strict verification procedures for both parties, ensuring that only trustworthy individuals have access to storage spaces.

6. Offer a Variety of Storage Options to Meet Diverse Needs

- Hamsaye caters to a wide range of storage needs, providing options for various types of belongings, from furniture and appliances to seasonal items and personal collectibles. This flexibility ensures that users can find the perfect storage solution for their unique circumstances.

7. Promote Transparency and User-Friendly Experience

- Hamsaye strives to create a transparent and user-friendly experience for both renters and storage providers. The platform provides detailed information about each storage option, including photos, descriptions, and pricing. Additionally, Hamsaye offers a support system to address any questions or concerns promptly.

8. Continuously Evolve and Adapt to Market Trends

- Hamsaye remains committed to innovation and adaptation, constantly exploring new ways to enhance its platform and meet the evolving needs of its users. This proactive approach ensures that Hamsaye remains a leader in the peer-to-peer storage space.

1.2 Scope

1. Target Users

- Hamsaye's primary target users are individuals and households seeking temporary or long-term storage solutions. The platform caters to a wide range of storage needs, including furniture, appliances, seasonal items, and personal collectibles.
- Renters: Individuals and households looking for convenient, affordable, and secure storage options in their local neighborhoods.
- Storage Providers: Individuals with extra storage space in their homes or garages who are interested in monetizing their unused assets.

2. Service Area

- Hamsaye initially operates in select urban and suburban areas, targeting communities with a strong need for convenient and affordable storage solutions.
- Expansion: As the platform grows, Hamsaye plans to expand its service area to cover a wider geographical range, reaching more neighborhoods and users.

3. Storage Options

- Hamsaye offers a variety of storage options to accommodate diverse needs and preferences, ranging from small lockers to large garages.
- Flexibility: Storage options are available for both short-term and long-term storage needs, catering to renters with varying storage requirements.

4. Platform Features

- Hamsaye provides a user-friendly online platform for both renters and storage providers. The platform facilitates easy browsing, searching, and booking of storage spaces.
- Communication Tools: The platform incorporates communication tools to enable renters and storage providers to connect and coordinate arrangements.
- Payment Processing: Hamsaye facilitates secure online payment processing for storage transactions.

5. Safety and Security

- Hamsaye prioritizes the safety and security of both renters and storage providers. The platform employs strict verification procedures for both parties, ensuring the trustworthiness of storage listings.

- Insurance Coverage: Hamsaye offers insurance coverage for stored belongings, providing peace of mind to renters.

6. Support Services

- Hamsaye provides a dedicated support system to address any questions, concerns, or issues that may arise during the storage process.
- Responsiveness: The support team is readily available to assist users promptly and efficiently.

7. Continuous Improvement

- Hamsaye commits to continuous improvement, constantly evaluating and enhancing its platform based on user feedback and market trends.
- Innovation: The platform explores new features and functionalities to enhance the user experience and maintain its competitive edge.

8. Community Engagement

- Hamsaye fosters a sense of community among its users, encouraging interactions and connections between renters and storage providers.
- Events and Initiatives: The platform organizes events and initiatives to promote community spirit and strengthen neighborhood ties.

1.3 Definitions, Acronyms, and Abbreviations

Definitions

- Hamsaye: A peer-to-peer storage platform that connects individuals with excess storage space with those who need temporary or long-term storage solutions.
- Renter: An individual or household seeking temporary or long-term storage solutions through Hamsaye.
- Storage Provider: An individual with extra storage space in their home or garage who lists their space on Hamsaye for rent.
- Peer-to-peer (P2P): A model of economic exchange in which individuals directly trade goods and services without the intervention of intermediaries.
- Convenience: The ease and accessibility of a storage solution, such as proximity to the renter's location and availability of online booking.
- Affordability: The cost-effectiveness of a storage solution in comparison to traditional self-storage facilities.
- Safety: The security of both renters' possessions and the storage spaces themselves.
- Security: The trustworthiness and credibility of storage providers.

- Transparency: The availability of clear and comprehensive information about storage options, pricing, and policies.
- User-friendly: The ease of navigation and use of the Hamsaye platform for both renters and storage providers.

Acronyms

- SRS: Software Requirements Specification

Abbreviations

- P2P: Peer-to-peer
- API: Application Programming Interface
- UI/UX: User Interface/User Experience

System Overview

2.1 System Description

Hamsaye is a peer-to-peer storage platform that connects individuals with excess storage space in their homes or garages (storage providers) with those who need temporary or long-term storage solutions (renters). The platform facilitates the seamless booking and management of storage spaces, providing renters with convenient and affordable storage options in their local neighborhoods.

2.2 System Architecture

Hamsaye's system architecture is a microservices architecture, which is a distributed architecture that breaks down a complex application into smaller, independent services. This architecture provides several benefits, including:

- Scalability: Microservices can be scaled independently, which allows Hamsaye to handle increasing demand without having to scale the entire application.
- Reliability: If one microservice fails, it does not bring down the entire application. Other microservices can continue to function, and the failed microservice can be restarted without affecting users.
- Maintainability: Microservices are easier to develop, test, and deploy than monolithic applications.

System Components

Hamsaye's system architecture consists of the following components:

- Frontend: The frontend is responsible for user interactions and is built using JavaScript, TypeScript, Webpack, Babel, Jest, Mocha, Chai, React.js, Axios, Redux, MobX, React Router, React Native, and Next.js.
- Backend: The backend manages the core functionalities of the platform and is built using Java, Spring boot, Spring Security, Spring Data JPA, Hibernate, OAuth2, JWT, and Slf4j. It communicates with the database using PostgreSQL and AWS.
- Database: The database stores all user information, storage listings, booking data, communication records, and insurance-related details. It is PostgreSQL, which is a relational database management system.
- Elasticsearch: Elasticsearch is a search and analytics engine that is used to index and search storage listings and user profiles.
- RabbitMQ: RabbitMQ is a message broker that is used to decouple microservices and enable asynchronous communication.
- Websockets: Websockets are used to provide real-time communication between renters and storage providers.

Requirements and Specifications

Frontend Requirements

- Languages: JavaScript, TypeScript
- Tools: Webpack, Babel, Jest, Mocha, Chai
- Libraries: React.js, Axios, Redux, MobX, React Router
- Frameworks: React Native, Next.js
- Functionalities:
    - User registration and authentication
    - Storage listing management
    - Search and filter functionality
    - Booking process
    - Real-time communication
    - Payment processing
    - Insurance coverage
    - Support system

Backend Requirements

- Languages: Java
- Libraries: Spring boot, Spring Security, Spring Data JPA, Hibernate, OAuth2, JWT, Slf4j
- Tools: PostgreSQL, AWS
- Functionalities:
    - Storage provider registration and verification
    - Storage listing creation and management
    - Payment processing
    - Communication tools
    - Insurance management

Database Requirements

- Type: Relational database
- Name: PostgreSQL
- Functionalities:
    - Store user information

- Store storage listings
- Store booking data
- Store communication records
- Store insurance-related details

2.3 System Functionalities

Hamsaye's core functionalities include:

- User Registration and Authentication: Renter and storage provider accounts are created and verified using secure authentication mechanisms.
- Storage Listing Management: Storage providers can create, edit, and remove their listings, providing detailed descriptions, photos, dimensions, pricing, and availability information.
- Search and Filter Functionality: Renters can search for storage listings using various criteria, including location, size, price, and availability.
- Booking Process: Renters can securely book storage spaces through the platform, specifying the duration and payment method.
- Real-time Communication: Renters and storage providers can communicate directly through the platform, exchanging messages and arranging logistics.
- Payment Processing: Hamsaye securely processes online payments for storage transactions, ensuring protection for both parties.
- Insurance Coverage: Hamsaye offers optional insurance coverage for stored belongings, providing renters with peace of mind.
- Support System: A dedicated support team is available to address user queries, concerns, and issues promptly.

2.4 System Integrations

Hamsaye integrates with third-party services to enhance its functionalities and user experience:

- Payment Gateway Integration: Secure integration with payment gateways for seamless online payment processing.
- Messaging Platforms Integration: Integration with messaging platforms like SMS, Bale, or email for real-time communication between renters and storage providers.
- Insurance Provider Integration: Integration with insurance providers to streamline insurance coverage processes and claims management.
- Location Services Integration: Integration with location services to provide accurate information on storage providers' locations and proximity to renters.

2.5 System Security

Hamsaye prioritizes security throughout its system by implementing robust measures:

- User Authentication and Authorization: Secure user authentication and authorization mechanisms to protect user accounts.
- Data Encryption: Storage of sensitive data, including user information and payment details, using industry-standard encryption protocols.
- Access Control: Restricted access to sensitive data and functionalities, ensuring that only authorized users can perform specific tasks.
- Vulnerability Monitoring: Continuous monitoring of the system for potential vulnerabilities and adopting timely security patches.
- Security Awareness Training: Regular security awareness training for employees to minimize human error and enhance overall security posture.

Functional Requirements

3.1 User Features

3.1.1 User Registration and Authentication:

- Allow both renters and storage providers to create accounts on the platform.
- Implement secure user registration and authentication procedures, including password hashing and multi-factor authentication.
- Store user information securely in the database, including name, email address, contact details, and payment information (if applicable).

3.1.2 Storage Listing Management:

- Enable storage providers to create, edit, and remove storage listings.
- Provide detailed listing information, including photos, descriptions, dimensions, pricing, availability, and location.
- Allow storage providers to filter listings based on various criteria, such as size, price, and location.

3.1.3 Search and Filter Functionality:

- Implement a comprehensive search function to allow renters to easily find storage listings that meet their specific needs.
- Provide filters for location, size, price, availability, and other relevant criteria.
- Offer suggestions and recommendations based on user search history and preferences.

3.1.4 Booking Process:

- Facilitate secure online booking of storage spaces for renters.
- Allow renters to select the desired storage listing, specify the duration of the booking, and choose their preferred payment method.
- Provide renters with a clear booking confirmation and detailed instructions for accessing the storage space.

3.1.5 Real-time Communication:

- Enable real-time communication between renters and storage providers through the platform.
- Provide a messaging feature for exchanging messages and coordinating arrangements.
- Integrate with messaging platforms like SMS, Bale, or email for additional communication options.

3.1.6 Payment Processing:

- Integrate with a secure payment gateway to process online payments for storage transactions.

- Handle secure payment processing for renters, ensuring the safety of their financial information.
- Provide clear and transparent payment options and pricing structures.

3.1.7 Insurance Coverage:

- Offer optional insurance coverage for stored belongings to renters.
- Provide renters with the option to select the level of coverage they require.
- Partner with reputable insurance providers to ensure adequate coverage for renters' belongings.

3.1.8 Support System:

- Establish a dedicated support system to address any questions, concerns, or issues that renters or storage providers may encounter.
- Provide prompt and efficient support through various channels, such as email, live chat, or phone calls.
- Establish a knowledge base with frequently asked questions and troubleshooting guides.

3.1.9 Additional Functionalities:

- Implement a ratings and reviews system to allow users to rate and review storage providers and listings.
- Integrate with location services to provide accurate information on storage providers' locations and proximity to renters.
- Enable storage providers to offer additional services, such as packing and unpacking assistance.
- Integrate with social media platforms to expand user reach and promote the platform.

By implementing these functional requirements, Hamsaye aims to provide a user-friendly and comprehensive platform that connects individuals with excess storage space with those who need temporary or long-term storage solutions. The platform's focus on convenience, affordability, safety, and security will empower individuals to make informed decisions about their storage needs while fostering a thriving community of renters and storage providers.

3.2 Administrative Features

3.2.1 User Management:

- Create, edit, and remove user accounts for both renters and storage providers.
- Enforce user verification processes, including identity verification and background checks, for storage providers.
- Manage user roles and permissions to control access to specific functionalities.

3.2.2 Data Management:

- Monitor and analyze user data to identify trends and improve platform performance.

- Maintain accurate and up-to-date information on storage listings, booking data, communication records, and insurance-related details.
- Implement data backup and recovery procedures to protect against data loss.

3.2.3 Reporting and Analytics:

- Generate comprehensive reports on platform usage, user demographics, and storage listing performance.
- Analyze data to identify areas for improvement and make informed business decisions.
- Provide reports to stakeholders and investors.

3.2.4 Payment Management:

- Process and manage online payments for storage transactions.
- Implement secure payment gateways and ensure compliance with payment processing regulations.
- Provide periodic reports on payment transactions and revenue generation.

3.2.5 Performance Monitoring:

- Monitor the performance of the platform to ensure stability and responsiveness.
- Identify and address any performance bottlenecks or errors.
- Measure key performance indicators (KPIs) to track the success of the platform.

3.2.6 Security and Compliance:

- Implement robust security measures to protect user data and prevent unauthorized access.
- Conduct regular security audits and vulnerability assessments.
- Comply with industry-standard security protocols and data privacy regulations.

3.2.7 Community Management:

- Foster a positive and supportive community among renters and storage providers.
- Address user feedback and resolve any disputes or complaints promptly.
- Promote responsible and ethical behavior among users.

3.2.8 Fraud Prevention:

- Implement measures to detect and prevent fraudulent activities, such as fake listings or unauthorized payments.
- Collaborate with partners to share fraud-related information and enhance detection capabilities.
- Maintain a transparent and accountable approach to fraud prevention.

3.2.9 Support and Maintenance:

- Provide ongoing support to renters and storage providers through various channels.

- Address technical issues promptly and efficiently.
- Regularly update and maintain the platform to ensure optimal performance and security.

3.3 Additional Features

3.3.1 Ratings and Reviews System:

- Implement a ratings and reviews system that allows renters to rate and review storage providers and listings.
- Provide a transparent and user-friendly interface for submitting and viewing ratings and reviews.
- Use ratings and reviews to provide insights into storage providers' reputations and the quality of their services.

3.3.2 Integration with Location Services:

- Integrate with location services to provide accurate information on storage providers' locations and proximity to renters.
- Allow renters to filter storage listings based on their location and proximity to their desired storage needs.
- Enhance the user experience by providing easy access to storage options in their local area.

3.3.3 Additional Services for Storage Providers:

- Enable storage providers to offer additional services, such as packing and unpacking assistance, to attract more renters.
- Create a marketplace for storage providers to sell packing supplies and other related products.
- Offer referral programs to incentivize storage providers to promote the platform to their networks.

3.3.4 Integration with Social Media Platforms:

- Integrate with social media platforms to expand user reach and promote the platform.
- Allow users to share storage listings, reviews, and experiences on social media.
- Leverage social media marketing strategies to attract new users and increase engagement.

3.3.5 Community Building Initiatives:

- Organize local events and gatherings for renters and storage providers to connect and build relationships.
- Create a forum or community space where users can discuss storage-related topics, share experiences, and seek advice.
- Partner with local businesses to offer discounts or promotions to Hamsaye users, further enhancing the platform's value proposition.

3.3.6 Additional Functionalities:

- Implement a feature to allow renters to request specific storage needs, such as climate-controlled or pet-friendly spaces.
- Develop a mobile application to provide a convenient and accessible platform for both renters and storage providers.
- Integrate with smart home devices to allow for remote access and control of storage units.

By implementing these additional features, Hamsaye aims to further enhance the user experience, promote community engagement, and differentiate itself from competitors. The platform's focus on convenience, affordability, safety, and security will continue to attract individuals seeking alternative storage solutions.

Non-functional Requirements

4.1 Performance Requirements

The Hamsaye platform must deliver a responsive and user-friendly experience across all user interactions, ensuring seamless navigation and prompt response times for both renters and storage providers. The platform must be equipped to handle a high volume of users and transactions simultaneously without compromising performance or experiencing latency. Moreover, the platform must possess scalability capabilities to accommodate future growth and demand, seamlessly adapting to increasing user base and transaction volume.

Performance Measures

1. Page loading and search query response times: The platform must consistently deliver rapid page load times and prompt responses to user searches, ensuring a smooth and efficient user experience.
2. Transaction processing time: The platform must handle storage booking and payment transactions efficiently, minimizing processing times and ensuring timely completion of user requests.
3. Concurrent user support: The platform must be able to support an optimal number of concurrent users without experiencing performance degradation or bottlenecks.
4. Scalability to handle increased traffic and usage: The platform must possess the ability to scale horizontally and vertically to accommodate future growth in user traffic and transaction volume.

4.2 Reliability Requirements

The Hamsaye platform must prioritize high availability and reliability, ensuring continuous operation and accessibility for its users. Robust systems must be implemented to prevent downtime, data loss, or disruptions. The platform must also demonstrate resilience and recover swiftly from any unforeseen incidents or failures.

Reliability Measures

1. Uptime and downtime metrics: The platform must maintain a high uptime percentage, minimizing downtime periods to a minimum acceptable level.
2. Mean time between failures (MTBF): The platform must exhibit a consistently long MTBF, indicating its ability to operate without frequent failures.
3. Mean time to recovery (MTTR): The platform must demonstrate a rapid MTTR, enabling prompt recovery from any disruptions or failures.
4. Disaster recovery plans and procedures: Comprehensive disaster recovery plans and procedures must be in place to ensure seamless resumption of operations in the event of major failures or disasters.

4.3 Security Requirements

The Hamsaye platform must maintain the highest standards of security to safeguard user data and protect against unauthorized access, disclosure, alteration, or destruction. The platform must adhere to all applicable data privacy regulations, such as GDPR and CCPA, and implement robust authentication and authorization mechanisms to control access to sensitive information. Additionally, industry-standard encryption protocols must be employed to protect sensitive data.

Security Measures

1. Secure user registration and login procedures: Implement strong authentication methods, including multi-factor authentication, to verify user identities and prevent unauthorized access.
2. Secure data storage and transmission: Employ industry-standard encryption protocols and secure data storage practices to safeguard sensitive user information during storage and transmission.
3. Data encryption for sensitive information: Encrypt all sensitive user data, such as payment information, to protect it from unauthorized access.
4. Regular security audits and vulnerability assessments: Conduct regular security audits and vulnerability assessments to identify and address potential security weaknesses or vulnerabilities.

4.4 Usability Requirements

The Hamsaye platform must prioritize user-friendliness and intuitiveness, ensuring that both renters and storage providers can easily navigate and utilize its functionalities effectively. The platform's user interface must be clear, intuitive, and consistent across all components, providing clear and concise instructions for all functionalities. Moreover, the platform must adhere to accessibility guidelines to accommodate users with disabilities.

Usability Measures

1. User satisfaction surveys: Conduct regular user satisfaction surveys to gather feedback regarding the platform's usability, identify areas for improvement, and track overall user experience.
2. User testing and usability studies: Perform user testing and usability studies to identify potential usability issues and refine the platform's interface and functionality.
3. Accessibility compliance testing: Conduct accessibility compliance testing to ensure that the platform is accessible to users with disabilities.

4.5 Compatibility Requirements

The Hamsaye platform must demonstrate broad compatibility with a wide range of devices, operating systems, and browsers to cater to a diverse user base. The platform must seamlessly function across desktops, laptops, tablets, and smartphones, while also accommodating various operating systems, such as Windows, macOS, iOS, and Android. Additionally, compatibility with various browsers, including Chrome, Firefox, Safari, and Edge, is crucial to ensure a consistent user experience.

Compatibility Measures

1. Cross-browser testing: Conduct thorough cross-browser testing to ensure that the platform's functionality and appearance are consistent across different browsers.
2. Cross-platform testing: Execute cross-platform testing to validate the platform's compatibility with various operating systems, ensuring seamless performance across devices.
3. Compatibility with assistive technologies: Implement compatibility with assistive technologies to ensure accessibility for users with disabilities.

Constraints

5.1 System Constraints

1. Scalability: The platform must be scalable to accommodate future growth in user traffic and transaction volume. This may require the ability to add additional servers, increase database capacity, or adopt cloud-based infrastructure.
2. Performance: The platform must maintain a high level of performance to ensure a responsive and user-friendly experience for both renters and storage providers. This may require optimizing code, implementing caching mechanisms, or using load balancing techniques.
3. Security: The platform must be highly secure to protect user data and prevent unauthorized access, disclosure, alteration, or destruction. This may require implementing robust authentication and authorization mechanisms, using industry-standard encryption protocols, and conducting regular security audits.
4. Usability: The platform must be easy to use for both renters and storage providers, regardless of their technical expertise. This may require creating a clear and intuitive user interface, providing clear and concise instructions, and conducting user testing and usability studies.
5. Compatibility: The platform must be compatible with a wide range of devices, operating systems, and browsers to cater to a diverse user base. This may require using cross-browser testing, cross-platform testing, and ensuring compatibility with assistive technologies.

5.2 Budget Constraints

1. Development Costs: The development of the Hamsaye platform will require a significant investment in software development resources, including developers, designers, and testers.
2. Infrastructure Costs: The platform will require hosting on a reliable cloud infrastructure, which will involve ongoing fees for server capacity, storage, and network connectivity.
3. Marketing Costs: To attract users to the platform, marketing expenses will be necessary for advertising, social media engagement, and content marketing.
4. Compliance Costs: The platform will need to comply with various data privacy regulations, such as GDPR and CCPA. This may involve legal fees, consultants, and ongoing compliance processes.
5. Maintenance Costs: The platform will require ongoing maintenance to address bugs, implement new features, and ensure security and performance.

By carefully considering these constraints, Hamsaye can prioritize its development and implementation strategies, ensuring a successful and sustainable platform that meets the needs of its users and achieves its business goals.

System Dependencies

6.1 External Dependencies

The Hamsaye platform relies on a range of external dependencies to function effectively and deliver a comprehensive user experience. These dependencies include:

1. Payment Gateway: The platform integrates with a secure payment gateway to process online payments for storage transactions. This ensures the protection of user financial information and compliance with payment processing regulations.
2. Messaging Platforms: The platform integrates with messaging platforms like SMS, Bale, or email to enable real-time communication between renters and storage providers. This facilitates seamless communication and coordination for booking arrangements and other inquiries.
3. Insurance Provider: The platform partners with a reputable insurance provider to offer optional insurance coverage for stored belongings to renters. This provides renters with peace of mind and protection against potential losses.
4. Location Services: The platform integrates with location services to provide accurate information on storage providers' locations and proximity to renters. This enables renters to filter storage options based on their location and preferences, enhancing the convenience of finding suitable storage solutions.
5. Social Media Platforms: The platform integrates with social media platforms to expand its reach and promote the platform to a wider audience. This allows users to share storage listings, reviews, and experiences on social media platforms, increasing platform visibility and engagement.
6. Data Storage Provider: The platform utilizes a reliable data storage provider to store user information, storage listings, booking data, communication records, and insurance-related details. This ensures the secure and efficient storage of sensitive data.
7. Content Delivery Network (CDN): The platform utilizes a CDN to distribute its content across multiple servers worldwide. This enhances website performance by reducing latency and ensuring faster page loading times for users worldwide.
8. Monitoring and Analytics Tools: The platform utilizes monitoring and analytics tools to track system performance, identify potential issues, and measure key performance indicators (KPIs). This allows for proactive maintenance and continuous optimization of the platform.
9. Third-Party APIs: The platform may integrate with various third-party APIs to provide additional features or functionalities. Examples include APIs for weather data, maps, or image processing.

By carefully managing these external dependencies, Hamsaye ensures that its platform remains integrated with reliable and reputable services, enhancing its overall functionality, security, and user experience.

6.2 Compatibility Dependencies

In addition to external dependencies, the Hamsaye platform also has compatibility dependencies to ensure a smooth user experience across various devices and operating systems. These compatibility dependencies include:

1. Web Browsers: The platform must be compatible with a wide range of web browsers, including Chrome, Firefox, Safari, and Edge. This ensures that users can access the platform and utilize its functionalities from their preferred browsers.
2. Operating Systems: The platform must be compatible with various operating systems, including Windows, macOS, iOS, and Android. This caters to a diverse user base and allows renters and storage providers to access the platform from their preferred devices.
3. Devices: The platform must be compatible with a variety of devices, including desktops, laptops, tablets, and smartphones. This ensures that users can access the platform from their preferred devices and enjoy a consistent user experience.
4. Device Resolutions: The platform's user interface must be responsive and adapt to different screen resolutions and device sizes. This ensures that users can comfortably interact with the platform, regardless of their device's display size.
5. Accessibility Features: The platform must adhere to accessibility guidelines and incorporate features that support users with disabilities. This includes features such as keyboard navigation, screen reader compatibility, and alternative text for images.

By carefully considering these compatibility dependencies, Hamsaye ensures that its platform is accessible and usable by users across a wide range of devices and operating systems, enhancing the overall user experience and inclusivity for all.

Documentation Requirements

7.1 User Manual

A comprehensive user manual is paramount for both renters and storage providers to effectively utilize the Hamsaye platform and fully comprehend its functionalities. The user manual should encompass the following aspects:

User Registration and Authentication:

- Convey clear instructions on creating a user account, including step-by-step guidance on completing registration forms and validating identities.
- Explain the distinct user roles (renters and storage providers) and the specific functionalities accessible to each role.

Storage Listing Management:

- Guide storage providers in creating, modifying, and managing storage listings. This includes providing instructions on uploading photos, providing comprehensive descriptions, establishing pricing, and managing availability.
- Illustrate how storage providers can filter and search for listings based on various criteria, such as location, size, pricing, and amenities.

Search and Filter Functionality:

- Assist renters in locating suitable storage solutions by explaining how to utilize the search bar and filters to narrow down their search results.
- Explain how renters can view in-depth information about specific storage listings, including photos, descriptions, pricing, and availability.

Booking Process:

- Mentor renters through the booking process, including selecting the desired storage listing, specifying the booking duration, and choosing their preferred payment method.
- Provide clear instructions on completing the booking process and receiving confirmation of their booking.

Real-time Communication:

- Explain how renters and storage providers can communicate via the platform employing a secure messaging feature.
- Provide guidelines on initiating conversations, sending messages, and coordinating booking arrangements.

Payment Processing:

- Illustrate how the platform integrates with a secure payment gateway to process online payments for storage transactions.
- Provide instructions on making payments for bookings and managing payment details on the platform.

Insurance Coverage:

- Explain the optional insurance coverage offered by Hamsaye and how renters can acquire coverage for their stored belongings.
- Guide renters through the process of selecting and purchasing insurance coverage.

Support System:

- Provide details on how renters and storage providers can access the platform's dedicated support system for assistance with any issues or questions.
- Explain the various support channels accessible, such as email, live chat, or phone calls.

Additional Features:

- Explain how renters can utilize ratings and reviews to assess storage providers and make informed decisions.
- Guide renters on integrating with location services to locate storage options near their desired locations.
- Illustrate how storage providers can offer additional services, such as packing and unpacking assistance, to attract more renters.

Integrations:

- Explain how the platform integrates with social media platforms to promote listings and gather user feedback.
- Describe the integration with third-party APIs, such as weather data or maps, to provide additional value to users.

7.2 Code Documentation Standards

Comprehensive and consistent code documentation is paramount for safeguarding the integrity, maintainability, and overall quality of the Hamsaye platform. To achieve this, the following code documentation standards must be rigorously adhered to:

Naming Conventions:

- Adopt Consistent Naming Conventions: Enforce a standardized naming convention for all code elements, including variables, functions, classes, and other constructs. This ensures uniformity and enhances code readability.

- Use Descriptive and Meaningful Names: Assign descriptive and meaningful names to code elements that accurately reflect their purpose and functionality. Avoid ambiguous, abbreviated, or cryptic names that could hinder comprehension.
- Avoid Redundancy: Minimize redundant comments that merely reiterate the code's obvious functionality. Comments should provide context, explain non-intuitive code sections, and document complex algorithms or unusual coding patterns.

Documentation Style:

- Adopt a Consistent and Clear Documentation Style: Establish a consistent documentation style throughout the codebase, ensuring clarity and consistency in comment formatting and syntax.
- Utilize Proper Indentation and Spacing: Employ proper indentation and spacing to enhance code readability and structure. Clear indentation allows for easy visual separation of code blocks, while adequate spacing improves code legibility.
- Format Comments Consistently: Adhere to a consistent style for commenting, using a unified syntax and format for all comment elements. This ensures uniformity and facilitates effortless code comprehension.

Code Structure:

- Implement Clean and Well-Structured Code: Adopt clean and well-structured code that adheres to best practices and industry conventions. Divide code into logical modules and functions to promote maintainability and enhance understandability.
- Make Code Self-Documenting: Employ descriptive naming conventions and meaningful comments to make the code self-documenting, minimizing the need for excessive comments. Well-structured code with clear naming should convey its purpose without extensive annotations.
- Utilize Documentation Tools Effectively: Leverage appropriate documentation tools, such as Javadoc or Doxygen, to generate comprehensive documentation from code comments. These tools can automate the process of generating documentation, freeing developers to focus on development tasks.

Documentation Tools Integration:

- Integrate Documentation Tools with Version Control Systems: Integrate documentation tools with version control systems, such as Git, to ensure that documentation updates are version-controlled and aligned with the corresponding code changes. This traceability facilitates seamless collaboration and prevents documentation divergence from the actual code implementation.

Code Review:

- Conduct Regular Code Reviews: Implement a rigorous code review process to identify and rectify any documentation gaps or inconsistencies. Code reviews provide an opportunity for peer scrutiny and ensure that documentation is kept up-to-date and aligned with the code.

- Foster a Culture of Code Documentation: Cultivate a culture of code documentation within the development team, emphasizing the importance of documenting code for future developers and maintenance. Encourage documentation as a natural part of the development process.
- Assign Documentation Responsibilities: Designate specific roles or individuals responsible for maintaining and updating code documentation. This ensures that documentation remains a priority and is consistently maintained.

Revision History

Revision Number	Date	Description	Author
1.0	2024-01-20	Initial draft of the System Requirements Specification (SRS)	Mohammad Hossein Alikhani
1.1	2024-01-21	Updated SRS with additional details on functional requirements, non-functional requirements, system dependencies, and documentation requirements and …	Mohammad Hossein Alikhani

Approval

9.1 Development Team Approval

Approval Date	Name	Role	Approval Comments
2024-01-20	Mohammad Hossein Alikhani	Founder and CEO	Approved the System Requirements Specification (SRS) for the Hamsaye Startup platform.
2024-01-20	Mohammad Mehdi Arab	Front-End Developer	Approved the SRS and provided feedback on the front-end aspects of the platform.
2024-01-20	Pouria Ghafarbeigi	Designer and Back-End Developer	Approved the SRS and provided feedback on the design and back-end aspects of the platform.

Appendix

10.1 Additional Information

Data Privacy and Security

The Hamsaye platform is committed to protecting the privacy and security of user data. The platform will implement robust security measures to safeguard user information, including:

- Secure user registration and authentication: Users will be required to create strong passwords and use two-factor authentication for additional security.
- Encrypted data storage: All user data will be encrypted at rest and in transit to prevent unauthorized access.
- Secure payment processing: The platform will integrate with a secure payment gateway to process online payments for storage transactions.
- Regular security audits: The platform will undergo regular security audits and penetration tests to identify and address any vulnerabilities.

Disaster Recovery

The Hamsaye platform will have a comprehensive disaster recovery plan in place to ensure business continuity in the event of an outage or data loss. The plan will include:

- Regular backups of user data: The platform will regularly back up user data to ensure that it can be restored in the event of an outage or data loss.
- Redundancy of data storage: The platform will use redundant data storage to ensure that user data is always available.
- Disaster recovery procedures: The platform will have a clear set of procedures for recovering from a disaster.

Accessibility

The Hamsaye platform will be designed and developed to be accessible to users with disabilities. This includes:

- Implementing WCAG 2.1 AA compliance: The platform will comply with the Web Content Accessibility Guidelines (WCAG) 2.1 AA, which are international standards for web accessibility.
- Using assistive technologies: The platform will be compatible with a variety of assistive technologies, such as screen readers and voice-based interfaces.
- Providing clear instructions: The platform will provide clear and concise instructions for all functionalities, making it easy for users to navigate and use the platform independently.

Future Enhancements

The Hamsaye platform is designed to be flexible and scalable, allowing for future enhancements and additions. Some potential enhancements include:

- Integration with additional location services: The platform could integrate with more location services to provide users with a wider range of storage options.
- Development of a mobile app: A mobile app would allow users to access the platform and manage their storage needs from their smartphones or tablets.
- Integration with social media platforms: The platform could further integrate with social media platforms to promote listings and gather user feedback.
- Integration with IoT devices: The platform could integrate with IoT devices to provide smart features, such as real-time occupancy tracking and temperature monitoring.