THE KEY COMPONENTS OF LUNARSPACE'S DECENTRALIZED ERP:

ZONES

Within this document is a detailed analysis of the key components of our systems. Please note: much of the content within this document may be subject to change.

TABLE OF CONTENTS

THE KEY COMPONENTS OF
LUNARSPACE'S DECENTRALIZED CRM:

ZONES

INTRODUCTION

Welcome to Lunarspace, where the cutting-edge world of web3 technology converges with the strategic imperatives of modern businesses. We are proud to present ZONES, our revolutionary platform, and the transformative $HAMP token, poised to reshape the way enterprises manage their operations, streamline processes, and unlock the full potential of blockchain integration.

In the digital era, businesses seek secure and decentralized data management, automation of sales and marketing efforts, and real-time tracking and rewarding of employee actions. With ZONES, these possibilities become a reality.

At Lunarspace, we are trailblazers in web3 business acceleration, driven by our commitment to redefining business operations with a robust ERP/CRM solution fortified by the unparalleled security and transparency of blockchain technology. ZONES, our flagship product, effortlessly integrates with cutting-edge web3 advancements, delivering unmatched security, automation, and access control.

Within this whitepaper, we delve deep into the technical intricacies of our platform, exploring the architecture of ZONES, the functionalities of the ZONES Launchpad and dPASS SBT, and the seamless implementation of the $HAMP protocol. Discover the boundless potential use cases and transformative benefits of ZONES, tailored to businesses of all sizes. With its customizable features, unwavering security, and user-friendly interface, ZONES emerges as the premier choice for businesses seeking to harness the full potential of web3 technology.

Join us on this journey as we unlock the true essence of ZONES and $HAMP, propelling enterprises of all kinds to new horizons of innovation and success. Embrace the future of web3 enterprise solutions with Lunarspace.

PROBLEMS SOLVED: BLOCKCHAIN IN ENTERPRISE

Problem #1: Complexity and lack of scalability. Many enterprise-level blockchain solutions are complex to set up and maintain, which can be a barrier for widespread adoption. Additionally, many blockchain networks struggle with scalability issues, which can limit their ability to handle large amounts of transactions and data.

Problem #2: Lack of interoperability. Different blockchain networks often have incompatible protocols and data formats, which can make it difficult for them to communicate and share data with one another. This can be a major obstacle for enterprise-level use cases, where multiple blockchain networks may be needed to support different business processes and systems.

Problem #3: Security concerns. One of the key concerns for enterprise-level blockchain usage is the security of the network. With the large amounts of sensitive data and financial transactions that take place on these networks, it is crucial that they are designed and implemented with robust security protocols in place.

ZONES addresses the problem of complexity and scalability by using the Cosmos SDK and Space and Time decentralized SQL databases, which provide a modular and easy-to-use framework for building and scaling blockchain applications. Additionally, ZONES' use of IBC allows for interoperability between different blockchain networks, enabling them to communicate and share data more easily.

In addition to maintaining traditional standards in network security protocols, ZONES also uses cutting-edge security protocols and methods such as the use of the dPASS SBT (a soulbound token) and the Keplr wallet integration. The utilization of these technologies is meant to provide an additional layer of security for enterprise-level blockchain usage, ensuring that sensitive data and financial transactions are always protected.

PROBLEMS SOLVED: WORKPLACE MANAGEMENT

Problem #1: Lack of integration with other systems and tools. Many organizations struggle with disparate systems that do not easily integrate with one another, making it difficult to get a holistic view of their operations.

ZONES addresses this issue by providing an integrated platform that can be easily connected to other systems, such as accounting software, e-commerce platforms, and marketing automation tools. This allows organizations to easily collect, store, and analyze data from multiple sources, providing them with a more complete view of their operations.

Problem #2: Inefficient processes and data silos. Many organizations struggle with manual, paper-based, or disjointed processes, resulting in massive inefficiencies.

ZONES addresses this issue by automating key processes and providing a decentralized repository for data, allowing organizations to streamline their operations and improve their bottom line. This is done by using cutting-edge technologies such as Space and Time decentralized SQL databases.

Problem #3: Insufficient use and lack of incentivization. One of the main challenges faced by businesses is how to encourage and incentivize employees to fully utilize their tools and systems.

The ZONES platform solves this problem by incorporating dynamic micro leaderboards and the $HAMP token. With dynamic micro leaderboards, employees can see their progress and compare their performance to that of their peers, fostering a sense of healthy competition and motivating them to achieve better results. The $HAMP token, on the other hand, rewards users with tokens for their contributions to the system. These incentives ensure that employees have a vested interest in utilizing the system to its fullest potential, which will lead to better overall performance.

WEB3 SOLUTIONS FOR WEB2 ENTERPRISES

ZONES is designed to be a comprehensive and secure enterprise resource planning (ERP) and customer relationship management (CRM) solution for businesses and organizations of all sizes. The platform is built using a combination of modern web technologies, including JavaScript, TypeScript, and Rust, as well as various APIs and open-source libraries.

One of the key features of the ZONES platform is its integration of the dPASS SBT and the ZONES Launchpad. The dPASS SBT is a soulbound token that is used to provide secure access to the ZONES platform. It is built using Cosmos SDK, which allows for the creation and management of a token on a blockchain. The token is soulbound, meaning it cannot be transferred, bought, or sold, and is only used for verifiable and controlled access to the platform.

WEB3 FEATURES

The ZONES Launchpad is a smart contract-based system that is used to manage the dPASS SBT and provide additional functionality for the ZONES platform. The Launchpad is built using the Cosmos SDK and is designed to be highly secure and modular, allowing for easy integration with other systems and protocols.

One of the key benefits of the ZONES platform is its ability to provide a comprehensive and secure solution for managing all aspects of a business or organization. The platform includes a wide range of tools and features for managing everything from financials and inventory to customer relationships and employee management. The integration of the dPASS SBT and the dERP Launchpad provides added benefits such as increased security and the ability to access and utilize new web3-based technologies and protocols.

In addition, the ZONES platform introduces two native tokens that play crucial roles within the ecosystem: $LSX and $HAMP.

$LSX is a CW20 smart contract token built on the Coreum blockchain, specifically designed to serve as the native token within the ZONES ecosystem. As the native token of ZONES, $LSX holds a pivotal position within the ecosystem, providing a range of functionalities, incentives, and benefits to its holders. It serves as a medium of exchange, a staking asset, and a representation of value within the ZONES platform. The issuance of $LSX tokens is limited, ensuring scarcity and value within the ecosystem.

$HAMP, on the other hand, is the Human Activity Mining Protocol within the ZONES platform. It is an internal tracking system that rewards users for contributing to the platform with customizable actions deemed valuable by a SuperAdmin. The SuperAdmin, usually a CEO, CTO, or Network Administrator, has full control over all accumulated $HAMP throughout their network, and it can be rewarded to their team as they see fit.

INTEGRATIONS

The integration of $LSX and $HAMP tokens within the ZONES platform provides additional benefits and incentives for users. $LSX can be staked in the pool with $CORE, the native token of Coreum, to earn additional rewards. Users can exchange their earned $HAMP tokens for $LSX at a fixed 1:1 ratio, unlocking the benefits and rewards associated with $LSX while leveraging the security and efficiency of Coreum.

In addition to the ZONES platform and its associated technologies, Lunarspace also offers a Keplr wallet integration, which allows for easy and secure management of digital assets within the platform such as the dPASS SBT, $HAMP, and $LSX, along with any other IBC-based assets.

The integration of $LSX and $HAMP tokens, along with the dPASS SBT and Keplr wallet, within the ZONES platform further enhances the web3 capabilities and features available to users. These tokens provide additional incentives, rewards, and opportunities for users to engage with the platform and benefit from its comprehensive and secure functionalities.

INTRODUCTION TO $HAMP

$HAMP is a fundamental component of the ZONES platform, designed to incentivize and reward user engagement, collaboration, and valuable contributions within the ecosystem. While we have touched upon $HAMP throughout the whitepaper, let us delve deeper into its role and significance within ZONES, as well as its relationship with Coreum ($CORE).

$HAMP, short for Human Activity Mining Protocol, is a Coreum Smart Token specifically created to reward users for their active participation and contributions to the ZONES platform. Through a process known as "human activity mining," users can earn $HAMP tokens by completing tasks, accomplishing goals, and engaging in meaningful activities within the platform.

The primary objective of $HAMP is to recognize and incentivize valuable human activity and foster a vibrant and engaged community within ZONES. By rewarding users for their contributions, $HAMP serves as a catalyst for collaboration, knowledge sharing, and overall productivity within the ecosystem.

The earned $HAMP tokens can be utilized in various ways within ZONES. Users can exchange their $HAMP for $LSX tokens, a CW20 smart contract built on Coreum, at a fixed 1:1 ratio. This provides them with a pathway to access the benefits and rewards associated with $LSX, while leveraging the security and efficiency of the Coreum blockchain.

Additionally, $HAMP tokens can be staked in the platform's decentralized treasury or used for specific activities within individual ZONES as determined by the SuperAdmin. This integration with Coreum allows for seamless interaction between $HAMP and $LSX, as well as other tokens and functionalities within the Coreum ecosystem.

The ability to earn and utilize $HAMP tokens offers tangible benefits to users. It not only enhances their engagement and interaction within ZONES but also provides an avenue for financial incentives and potential long-term value. The decentralized and transparent nature of the $HAMP protocol, built on Coreum, ensures that users' contributions are duly recognized, fostering a sense of ownership and empowerment within the platform.

Furthermore, the distribution and allocation of $HAMP tokens within ZONES are governed by predefined rules and algorithms, leveraging the consensus mechanisms of Coreum. This ensures fairness and consistency in the reward system, allowing users to trust the integrity and credibility of both $HAMP and Coreum.

In summary, $HAMP tokens, as a Coreum Smart Token, play a vital role in incentivizing, rewarding, and fostering a thriving ecosystem within ZONES. They recognize and value user contributions, fuel collaboration, and drive engagement. By integrating with Coreum, $HAMP establishes a strong connection to the broader blockchain ecosystem, enabling seamless interactions with $LSX and other tokens within the Coreum network.

By introducing the relationship between $HAMP, $LSX, and Coreum early in the whitepaper, readers gain a comprehensive understanding of how these components work together to incentivize user activity and create a robust and interconnected ecosystem within ZONES.

INTRODUCTION TO $LSX

$LSX is a CW20 smart contract token built on the Coreum blockchain, specifically designed to serve as the native token within the ZONES ecosystem. While we have discussed $LSX earlier in the whitepaper, let us now explore its role and relationship with Coreum ($CORE) in more detail.

As the native token of ZONES, $LSX holds a pivotal position within the ecosystem, providing a range of functionalities, incentives, and benefits to its holders. It serves as a medium of exchange, a staking asset, and a representation of value within the ZONES platform.

The issuance of $LSX tokens is limited, ensuring scarcity and value within the ecosystem. This scarcity contributes to the desirability and long-term potential of $LSX, making it an attractive asset for users and investors alike.

One of the significant features of $LSX is its relationship with $HAMP, the Coreum Smart Token within the ZONES platform. Users can exchange their earned $HAMP tokens for $LSX at a fixed 1:1 ratio. This allows users to unlock the benefits and rewards associated with $LSX, while leveraging the security and efficiency of Coreum.

$LSX tokens can be staked in the pool with $CORE, the native token of Coreum, to earn additional rewards. By participating in staking, users contribute to the security and stability of the Coreum blockchain while being rewarded for their commitment and participation.

Moreover, the decentralized treasury of each Zone within the ZONES platform is funded by the SuperAdmin and utilizes $HAMP tokens. These $HAMP tokens can be exchanged for $LSX, providing liquidity to the treasury and incentivizing users to earn and engage with $HAMP.

The integration of $LSX with Coreum ensures seamless interoperability and connectivity within the broader blockchain ecosystem. Users can leverage the advantages of Coreum's decentralized infrastructure, such as fast and secure transactions, while enjoying the benefits of $LSX within the ZONES platform.

By holding $LSX, users gain access to various privileges and features within ZONES, including voting rights for governance decisions, exclusive rewards and bonuses, and potential participation in future token-related events or offerings. $LSX serves as a representation of active involvement and commitment within the ZONES community.

In summary, $LSX, as a CW20 smart contract token built on Coreum, plays a vital role within the ZONES ecosystem. It serves as the native token, enabling a wide range of functionalities, incentives, and benefits. The seamless integration with Coreum allows users to leverage the advantages of both $LSX and Coreum, fostering a robust and interconnected ecosystem within ZONES.

By outlining the relationship between $LSX, $HAMP, and Coreum, readers gain a comprehensive understanding of how these tokens work together to incentivize user participation, provide liquidity, and foster a thriving ecosystem within ZONES.

WEB3 USE CASES

ZONES offers a wide range of web3 use cases that are designed to help businesses and organizations take advantage of the benefits of blockchain technology. These use cases leverage the integration of the native tokens, $LSX and $HAMP, within the ZONES ecosystem, along with other web3 features and technologies.

One of the main use cases is the integration of the Human Activity Mining Protocol ($HAMP), which allows organizations to reward their users for valuable activities on the platform. By utilizing $HAMP, organizations can incentivize desired behaviors and increase user engagement within the ecosystem. Users can earn $HAMP for contributing to the platform, and the accumulated tokens can be rewarded by the SuperAdmin to their team or exchanged for $LSX, the native token of ZONES. This not only encourages active participation but also provides organizations with a new revenue stream through the sale of $HAMP tokens.

Another web3 use case is the integration of the dPASS SBT, a soulbound token, which enhances security and controlled access to ZONES. By implementing a token-based access control system, organizations can ensure that only authorized individuals have access to their sensitive data and systems. This adds an additional layer of security, particularly crucial for organizations that handle sensitive information. The integration of the dPASS SBT provides a secure and controlled environment for accessing and utilizing the features of ZONES.

The ZONES Launchpad is another web3 feature that allows organizations to easily create and manage smart contracts on-chain. By utilizing the Launchpad, organizations can automate complex business processes, such as supply chain management, and reduce the need for manual intervention. The user-friendly interface of the ZONES Launchpad simplifies the deployment and interaction with token utility, making it accessible even to non-technical users. The Launchpad enables organizations to leverage the power of smart contracts and decentralized technologies within their operations.

Finally, the integration of the Keplr wallet further enhances the web3 ecosystem within ZONES. The Keplr wallet provides users with a secure and convenient solution for storing and managing their digital assets. By utilizing the Keplr wallet, users can securely interact with decentralized applications, perform transactions on-chain, and seamlessly access the web3 ecosystem. This integration empowers users to fully participate in the benefits of blockchain technology within the ZONES platform.

Overall, ZONES offers a wide range of web3 use cases that can help organizations streamline their operations, increase efficiency, and security while improving their bottom line. By leveraging the power of blockchain technology, organizations can gain a competitive advantage in an increasingly digital world. The integration of $LSX and $HAMP tokens, along with other web3 features, provides organizations with the tools and capabilities to unlock the potential of decentralized technologies and revolutionize the way they operate.

GETTING TECHNICAL

The next section of the whitepaper will provide a detailed overview of the products and protocols offered by Lunarspace. This will include an in-depth description of the ZONES platform, the ZONES Launchpad, the dPASS SBT, Keplr wallet integration, and any other relevant technologies or features.

First, we will discuss ZONES; a robust ERP/CRM platform that provides businesses with a streamlined and efficient way to manage their operations. The platform offers a wide range of features, including customer relationship management, inventory management, financial management, and more. Additionally, ZONES integrates with web3 technologies such as $HAMP, $LSX, the ZONES Leaderboards, Space and Time decentralized SQL databases, and the dPASS SBT and ZONES Launchpad. These features allow for a more decentralized, transparent, and useful system for businesses of all kinds.

A BRIEF REFRESHER

The ZONES Launchpad is a smart contract-based protocol that enables businesses to launch, manage, and govern their own tokens within the Cosmos Hub. This launchpad, powered by the Cosmos SDK and integrated with Coreum, is designed to be easy to use, secure, and efficient. It provides businesses with the capability to create and manage their own tokens, including the native token of ZONES, $LSX, and the Human Activity Mining Protocol token, $HAMP.

The dPASS SBT is a soulbound token that plays a vital role in ZONES by granting secure access to the ZONES platform. This token cannot be transferred, bought, or sold, and is exclusively awarded to users by the SuperAdmin. By utilizing the dPASS SBT, ZONES ensures that only authorized users have access to the system and sensitive data, providing an additional layer of security and controlled access.

The Keplr wallet integration enhances the user experience by seamlessly connecting users' Keplr wallets to the ZONES platform within the Launchpad. This integration enables users to securely manage their digital assets, including the dPASS SBT, $HAMP, and other tokens from the Cosmos network. With the Keplr wallet integration, users can easily perform various actions such as transferring tokens or participating in staking.

In addition, ZONES incorporates a dynamic leaderboard feature that serves as an incentive for user engagement and contributions. The leaderboard utilizes the Human Activity Mining Protocol ($HAMP) to reward users for their valuable contributions and engagement within the platform. By utilizing the leaderboard and the $HAMP token, businesses can acknowledge and reward their top contributors, while users are motivated to compete and strive for rewards.

ZONES

Lunarspace's ZONES is a robust and secure CRM/ERP platform that is built on top of cutting-edge web3 technologies. The platform is built primarily using Angular13 and Rust for their high performance, design, and security capabilities and utilizes TypeScript for its front-end development. The platform also makes use of various APIs and libraries such as the Cosmos SDK and the Tendermint Protocol to interact with the blockchain and ensure secure and decentralized access control.

At its core, ZONES is designed to provide businesses with a single, unified platform for managing all their customers and internal business data. The platform includes a wide range of features and functionalities that allow businesses to manage their customer relationships, sales, marketing, and operations all in one place.

One of the key features of ZONES is its ability to securely manage and store large amounts of customer and internal business data in a decentralized manner. With Space and Time decentralized SQL databases, the platform uses advanced encryption and security protocols to ensure that all data is protected from unauthorized access and breaches. Additionally, the platform is designed to be fully compliant with all relevant data privacy regulations, such as GDPR and CCPA.

STREAMLINING EFFICIENCY

ZONES includes a powerful set of analytics and reporting tools that allow businesses to gain valuable insights into their customer and business data. These tools allow businesses to track key metrics such as customer acquisition and retention, sales performance, marketing ROI, and more.

ZONES also includes a built-in workflow management system that allows businesses to automate and streamline their internal processes. This allows businesses to manage tasks more efficiently such as sales lead follow-up, customer service, and more.

In addition to these core features, ZONES also includes several advanced web3-based features such as the integration of Human Activity Mining Protocol (HAMP) and the leaderboards. These features allow businesses to track and reward their team's performance, enabling efficient utilization of the platform.

These components work together to provide a comprehensive and secure ERP/CRM platform that is fully integrated with web3 technologies. By using smart contracts and digital assets, the platform can automate processes, enforce business rules, and provide a secure means of access control. With the added benefits of $HAMP and the leaderboards, ZONES is a powerful solution that can help businesses manage their internal processes and improve their bottom line.

ZONES: KEY FEATURES

ZONES, the flagship product of Lunarspace, is a robust and feature-rich ERP/CRM platform that is built using a combination of modern technologies and web3 integration.

One of the key features of ZONES is its integration with web3 technologies, enabling businesses and organizations to leverage the benefits of blockchain technology. This includes the use of soulbound tokens, such as the dPASS SBT, which provides secure access to the ZONES platform and ensures that only authorized users have access to sensitive information. Additionally, ZONES also integrates with the Keplr wallet, which allows users to securely manage their digital assets and access the platform using their wallet credentials.

ZONES is designed to be modular and customizable, which allows businesses and organizations to tailor the platform to their specific needs. This includes features such as the ability to create custom workflows, configure access controls, and integrate with external systems and databases. Another important component of ZONES is the use of Space and Time decentralized SQL databases. This allows for the storage and retrieval of data in a decentralized and distributed manner, which provides several benefits such as improved security, scalability, and availability.

In terms of operational functionalities, ZONES offers a wide range of features that are designed to help businesses and organizations streamline their operations and improve their bottom line. These include:

- Contact management: ZONES allows users to manage and organize their contacts, including the ability to segment them into different groups and lists.
- Sales management: ZONES includes a comprehensive sales management module that allows users to track leads, opportunities, and deals, as well as manage their sales pipeline.
- Marketing automation: ZONES includes a powerful marketing automation module that allows users to create and execute marketing campaigns, track their performance, and measure their ROI.
- CRM: ZONES includes a comprehensive CRM module that allows users to manage their customer relationships, including the ability to track customer interactions, manage customer data, and create custom reports.
- Project management: ZONES includes a project management module that allows users to create, track, and manage projects, assign tasks, set deadlines and view progress.
- Time tracking: ZONES allows users to track time spent on projects, tasks, and activities, as well as generating detailed reports on time usage.
- Accounting: ZONES includes an accounting module that allows users to manage their financials, including invoicing, expenses, and reporting.

- Human Resource Management: ZONES includes an HRM module that allows users to manage employee information, track attendance, manage payroll and benefits, and handle employee performance evaluations.
- Reporting and Analytics: ZONES includes a reporting and analytics module that allows users to create custom reports and analyze their business data in real-time.
- Integration: ZONES can integrate with other systems and software that has open API to provide a seamless experience for users.

Additionally, ZONES also offers robust security features, including user access control, encryption, and secure, decentralized data storage to ensure that client's data is always protected. With the integration of $HAMP and the dPASS SBT, ZONES offers an added layer of security and access control to ensure that only authorized users have access to the platform.

ZONES LAUNCHPAD

The ZONES Launchpad is a dApp, but to be more exact, it is a contract-based access control system built on top of the Juno Network. It allows for the creation and management of users and their corresponding roles within the dERP platform. It uses JSON-Schema to define the contract's structure and input/output parameters for each function and is implemented using Rust and TypeScript.

The contract includes several message types for performing various actions such as creating and archiving users, changing the owner of the contract, and setting the address of the dPASS SBT contract. The contract also includes several query types for querying the state of the contract and the users it manages.

The contract uses the Cosmos SDK's built-in access control capabilities to ensure that only authorized parties can perform actions on the contract and gain access to its data. It is designed to be upgradeable, allowing for future updates and improvements to be implemented without affecting the existing functionality.

The ZONES Launchpad works with the browser based Keplr Wallet extension to provide users with an easy and secure way to access their platform account. The dPASS SBT is used to bind the user to their account and provide them with secure access to the platform. The ZONES Launchpad interacts with the dPASS SBT contract to ensure that only authorized users can access the platform and perform actions.

dPASS SBT

The dPASS SBT is a unique token that is used for secure access to the ZONES platform. It is built using the Cosmos SDK, a modular blockchain development framework, and is designed to be soulbound, which means that it cannot be transferred, bought, or sold. The dPASS is created and managed using the dERP SBT Contract, which is a smart contract that is deployed on the Juno Network.

The dPASS SBT is created during the instantiation of the dERP SBT Contract, where the contract's admin, name and symbol are set. The admin of the contract is responsible for adding and removing users as well as minting new tokens. Only admins and the SuperAdmin can mint new tokens, and they are only minted to users that have been added to the whitelist by said admin or SuperAdmin.

The dPASS SBT uses a token ID to identify each token, which is a unique 64-bit unsigned integer that is assigned to the token during the minting process. The token ID is used to query the token's owner and to track the token's movements on-chain.

To ensure the security of the dPASS SBT, the contract uses the Cosmos SDK's built-in access control features, such as the ability to add and remove admins and users and to check the token's ownership. This ensures that only authorized users can mint and transfer tokens, and that the token's ownership can be audited at any time.

Overall, the dPASS SBT is an extremely crucial component of the ZONES platform, as it provides a secure and auditable way to manage access to the platform and to track the movements of your tokens on-chain.

dERP BASE CONTRACT & dERP SBT CONTRACT

(ZONES Launchpad and the dPASS SBT)

The dERP Base Contract and the dERP SBT Contract are smart contracts that are used to manage and interact with the ZONES platform. Both contracts use the JSON-Schema to define the contract's structure and the input/output parameters for each function. The dERP Base Contract uses Rust, TypeScript and Shell code, in order to provide the necessary functionality for the platform.

The dERP Base Contract has an instantiate function that takes in several required parameters such as the denomination of the token, the ID of the contract, the owner of the contract, the SBT address of the contract, and the URL of the contract. The execute function of the dERP Base Contract allows for several different actions to be taken, such as minting a new SBT, setting the SBT address, changing the owner, creating a user, archiving a user, unarchiving a user, enabling admin access, disabling admin access, adding a minter, and removing a minter. Each of these actions takes in specific required parameters, such as the address of the user for actions like creating a user or changing the owner.

The dERP SBT Contract also has an instantiate function that takes in three required parameters: the admin of the SBT, the name of the SBT. and the symbol of the SBT. The execute function of the dERP SBT Contract allows for several different actions to be taken, such as minting new SBT, adding a minter, and removing a minter. The mint action takes in an owner and **'token_uri'** as required parameters. The **'add_minter'** and **'remove_minter'** actions take in the minter address as a required parameter.

The query function of the dERP SBT Contract has several query options, such as **'holds_sbt'**, **'owner_of'**, **'total_supply'** and **'metadata_of'** – each of which returns different information about the SBT. The **'holds_sbt'** query takes in an address as a parameter and returns a boolean indicating if the address holds any of the SBT. The **'owner_of'** query takes in a token ID as a parameter and returns the address of the owner of that specific token. The **'total_supply'** query returns the total number of tokens currently in circulation. The **'metadata_of'** query takes in a token ID as a parameter and returns the metadata associated with that specific token.

Both contracts use the Cosmos SDK, a modular blockchain development framework, to perform the necessary functions and interactions with the blockchain. The Cosmos SDK provides a set of modules that can be used to perform various functions such as governance, staking, and token transfer. The dERP Base Contract and dERP SBT Contract both utilize the Cosmos SDK to interact with the blockchain and perform the necessary actions.

MANAGING RESPONSIBILITIES

The dERP Base Contract is responsible for managing the creation and management of users, organizations, and SBT addresses. It defines several message types that are used to perform various actions such as minting a new SBT, setting the SBT address for a user, changing the owner of the contract, creating new users, archiving and unarchiving users, and enabling and disabling admin privileges for users. Each message type is defined using JSON-Schema which ensures that the input parameters are in the correct format and that all required fields are present.

The instantiate message type is used to create a new instance of the contract and requires the following fields: **"denom"**, **"id"**, **"owner"**, **"sbt_address"**, and **"url"**. The **"denom"** field is the denomination of the SBT, the **"id"** field is the unique identifier for the contract, the **"owner"** field is the address of the owner of the contract, the **"sbt_address"** field is the address of the SBT, and the **"url"** field is the URL of the contract's metadata.

The execute message type is used to perform various actions on the contract and is defined as a **"oneOf"** object, meaning that it can take one of several different forms depending on the action being performed. The **'mint_sbt'** message type is used to mint a new SBT and takes a single object as input. The **'set_sbt_address'** message type is used to set the SBT address for a user and takes an object with a single field **"addr"** which is the address of the SBT. The **'change_owner'** message type is used to change the owner of the contract and takes an object with a single field **"addr"** which is the address of the new owner. The **'create_user'** message type is used to create a new user and takes an object with a single field **"user_addr"** which is the address of the user.

The **'archive_user'** message type is used to archive a user, which means that the user will no longer have access to the ZONES platform. The message type requires a single parameter, **"addr"**, which is the address of the user to be archived. The **'unarchive_user'** message type is used to unarchive a user, which means that the user will regain access to the ZONES platform. This message type also requires a single parameter, **"addr"**, which is the address of the user to be unarchived.

Both the **'archive_user'** and **'unarchive_user'** message types are typically used by the SuperAdmin or other authorized users to manage access to the ZONES platform for specific users.

Additionally, the **'enable_admin'** message type is used to enable admin access for a user, which means that the user will have access to the ZONES platform with admin privileges. This message type also requires a single parameter, **"addr"**, which is the address of the user to be granted admin access.

All of these message types, along with the other available message types in the dERP Base Contract, work in conjunction to allow for the management and control of users, access, and privileges within the ZONES platform.

TOKEN MINTING

The dERP SBT Contract, on the other hand, is used to manage the creation, minting and control of the dPASS SBTs. The contract uses the mint message to mint a new dERP SBT token, which requires the input of the owner of the token, and an optional input of **'token_uri'**. The **'add_minter'** message is used to add a minter to the contract, which is a user who is authorized to mint new dPASS SBTs. This message requires the input of the minter which is the **"address"** of the user who will be added as minter. The **'remove_minter'** message is used to remove a minter from the contract, which is a user who is no longer authorized to mint new dPASS SBTs. This message also requires the **"address"** of the user to be passed as a parameter in the object.

The **'enable_admin'** message type is used to grant admin privileges to a user. This message also requires the **"address"** of the user to be passed as a parameter in the object.

The dERP SBT Contract, on the other hand, is focused on the management of the dPASS SBT. The instantiate message type takes in three required parameters: admin, name and symbol. The admin parameter is a string that represents the admin address of the SBT, the name parameter is a string that represents the name of the SBT, and the symbol parameter is a string that represents the symbol of the SBT.

The execute message type in the dERP SBT Contract contains three options: **'mint'**, **'add_minter'** and **'remove_minter'**. The mint message type is used to mint new tokens and requires an object with the owner parameter, which is a string representing the address of the token owner and an optional **'token_uri'** parameter, which is a string representing the Universal Resource Identifier (URI) of the token. The **'add_minter'** message type is used to add a minter address and requires an object with the minter parameter, which is a string representing the minter's address. The **'remove_minter'** message type is used to remove a minter address and also requires an object with the minter parameter, which is a string representing the minter's address.

The query message type in the dERP SBT Contract contains two options: **'holds_sbt'** and **'owner_of'**. The **'holds_sbt'** message type is used to check if an address holds the certain utility of the dPASS SBT and requires an object with the **'addr'** parameter, which is a string representing the address. The **'owner_of'** message type is used to check the owner of a certain token ID and requires an object with the **'token_id'** parameter, which is an integer representing the token ID.

Both contracts also include a query function that allows for querying certain information from the contract. The dERP Base Contract includes the **"holds_sbt"** query which takes an **"addr"** parameter, which is the address of the user to check if they are authorized to use the platform and returns a boolean indicating if they are or not. The **"owner_of"** query takes a **"token_id"** parameter, which is the unique identifier of the dPASS SBT, and returns the address of the owner of that token. The dERP SBT Contract includes the **"total_supply"** query which returns the total number of tokens that have been minted. The **"balance_of"** query takes an **"addr"** parameter, which is the address of a user, and returns the number of tokens that user holds. The **"owner_of"** query takes a **"token_id"** parameter, which is the unique identifier of a token, and returns the address of the owner of that token.

LANGUAGE FUNCTIONALITY

Both contracts are written in Rust, a systems programming language that runs blazingly fast, prevents segfaults, and guarantees thread safety, it is also the language which is used by the Cosmos-SDK. TypeScript, a typed superset of JavaScript that compiles to plain JavaScript, is used for the front-end and some of the back-end logic. The contracts use the Cosmos SDK, a modular blockchain development framework, to perform the necessary functions and interactions with the blockchain. The dERP Base Contract is also using the Keplr wallet implementation, dPASS SBT instantiation, and authentication for access control.

The dERP SBT Contract uses the Cosmos SDK to perform the necessary functions related to creating and managing a new token. It has an **"instantiate"** message type, which creates a new token with the details provided in the **"MintMsg"** object. This object requires the **"admin"**, **"name"**, and **"symbol"** fields to be provided, with **"admin"** representing the admin of the token, **"name"** representing the name of the token, and **"symbol"** representing the symbol of the token.

The **"execute"** message type of the dERP SBT Contract has several options for interacting with the token. The **"mint"** option allows for the minting of new tokens, with the **"MintMsg"** object required as the input. This object requires the **"owner"** field to be provided, which represents the owner of the newly minted token. Additionally, it includes an optional **"token_uri"** field, which is a universal resource identifier for this token and should point to a JSON file that conforms to the CW721 Metadata JSON Schema.

Another option for the **"execute"** message type is **"add_minter"** and **"remove_minter"** which are used to add or remove minters to the token. Both options require "minter" field to be provided which is the address of the minter.

Finally, the **"query"** message type of the dERP SBT Contract allows for querying information about the token, such as the owner of a specific token (using the **"owner_of"** message type) or if an address holds a specific token (using the **"holds_sbt"** message type).

In summary, the dERP SBT Contract is using the Cosmos SDK to create and manage a token, with functionalities such as minting new tokens, adding, and removing minters, and querying information about token holders and token ownership. The contract defines several message types for interacting with the contract, including the **"mint"** message type for minting new tokens, **"add_minter"** and **"remove_minter"** message types for adding and removing minters, and **"holds_sbt"** and **"owner_of"** message types for querying information about token holders and ownership. The contract also uses JSON-Schema to define the structure and input/output parameters for each function, ensuring the data passed to the contract is in the correct format.

$HAMP AND $LSX BUILT ON COREUM

$HAMP (Human Activity Mining Protocol) and $LSX (Lunarspace Token) are tokens that support the ZONES ecosystem and its users. Built on the Coreum blockchain using CosmWasm and the Cosmos SDK, $LSX is the main currency within the ZONES ecosystem, earned by users for performing various activities. $HAMP, a Coreum smart token, is used to reward users with $LSX for using the platform efficiently. The SuperAdmin (owner, CTO, or Network Admin) of a Zone controls the rewards and can stake $LSX in the Zone community pool. $HAMP is a customizable smart token unique to each Zone within ZONES, earned by performing activities within the platform and swapped for $LSX at a 1:1 ratio.

Both tokens have various features and functionalities implemented using the CosmWasm framework and are designed to incentivize activities within the platform and reward users accordingly. The use of these tokens brings benefits to enterprises, such as increased participation and engagement among employees, as well as benefits to the blockchain community through increased adoption and use of both the Coreum and Cosmos ecosystems.

USE CASES FOR $LSX

1. Incentivizing activities: $LSX is used to incentivize and reward users for performing various activities within the ZONES platform, such as completing tasks, contributing content, and engaging with other users.

2. Token swap: $HAMP can be swapped for $LSX at a 1:1 ratio, enabling users to participate in the activities of different ZONES within the ZONES ecosystem.

3. Staking: $LSX can be staked in the Zone community pool (decentralized treasury), where users can earn additional rewards by providing liquidity for staking.

4. Exchanging for $CORE: $LSX can be exchanged for $CORE, the native cryptocurrency of the Coreum network, which can be used for transactions and other activities within the wider blockchain ecosystem.

5. Leaderboard: A leaderboard can be implemented to display the top earners of $LSX rewards within the organization, incentivizing users to earn more rewards and increasing engagement with the platform.

USE CASES FOR $HAMP

1. Incentivizing Activities: $HAMP tokens are earned by users for performing various activities within the platform, such as completing tasks or contributing to projects. This incentivizes users to participate and engage more actively within the ZONES ecosystem.

2. Customizable Rewards: Each Zone within ZONES can customize their $HAMP token to fit their specific needs and preferences. This allows for greater flexibility and adaptability in incentivizing activities and rewarding users.

3. Token Swap: $HAMP tokens can be swapped for $LSX tokens at a 1:1 ratio, allowing users to convert their earned tokens into the main currency within the ZONES ecosystem.

4. Increased Adoption: By using $HAMP tokens within the ZONES platform, enterprises can increase adoption and use of the Coreum and Cosmos ecosystems, which can in turn benefit the broader blockchain community.

Both $HAMP and $LSX have several features and functionalities that are implemented using the CosmWasm framework within Coreum:

- Issuance
- Access Control
- Burning
- Freezing and Global Freezing
- Whitelisting
- Blacklisting
- Burn Rate

- Send Fee
- IBC Compatibility
- Smart Contract Integration
- Token Swap
- Staking
- Cash-out
- Leaderboard

$HAMP: A SMART TOKEN ON COREUM

$HAMP (Human Activity Mining Protocol) is a unique smart token built on the Coreum blockchain, leveraging the advanced capabilities of Coreum's smart contract platform. As a smart token, $HAMP introduces a new paradigm for incentivizing and rewarding user engagement and activity within the ZONES ecosystem.

Utilizing the power of smart contracts, $HAMP offers a customizable and programmable token that enables ZONES to design and implement their own unique reward systems tailored to their specific needs. This flexibility allows SuperAdmins to align the reward structure with their organization's goals and desired behaviors, fostering a culture of productivity, collaboration, and innovation.

The Coreum blockchain, known for its robustness and scalability, serves as the perfect foundation for $HAMP's implementation. Built using Coreum's smart contract language, the token benefits from the platform's security, transparency, and immutability. This ensures that rewards and transactions within the ZONES ecosystem are executed with utmost integrity and trust.

With $HAMP, the concept of "work-to-earn" is brought to life. Users are incentivized to actively participate and contribute within the ZONES platform, earning $HAMP tokens as a result. These tokens can be accumulated, redeemed, or exchanged for $LSX, the primary currency within the ZONES ecosystem, at a 1:1 ratio.

The smart contract underlying $HAMP allows for the implementation of a wide range of features and functionalities. SuperAdmins have granular control over the token, enabling them to define the token issuance rate, customize reward structures, set access control rules, and establish governance parameters. This level of configurability empowers SuperAdmins to create dynamic and engaging reward systems that align with their organization's objectives and foster desired behaviors.

The transparency and immutability of Coreum's smart contracts ensure that $HAMP rewards are allocated and distributed fairly. Every transaction and reward is recorded on the blockchain, providing users with full visibility and auditability of their earnings. This transparency enhances trust among users, as they can verify the accuracy and legitimacy of the rewards they receive.

By utilizing $HAMP as a smart token on Coreum, the ZONES ecosystem unlocks a powerful tool for incentivizing and rewarding user engagement. It paves the way for a new era of decentralized work systems, where individuals are recognized and rewarded for their contributions. Through the seamless integration of smart contracts and blockchain technology, $HAMP transforms the way organizations motivate and engage their workforce, driving innovation and efficiency within the ZONES ecosystem.

TOKENOMICS

The tokenomics of $LSX and $HAMP are designed to incentivize activity within the ZONES ecosystem. For example, when an employee completes a task or contributes to the community, they are rewarded with $LSX. The more $LSX they earn, the more they can stake in the pool with $CORE, which can then earn them even more rewards. Additionally, the SuperAdmin of each Zone can incentivize specific activities within their Zone by rewarding employees with $HAMP.

The tokenomics for $LSX and $HAMP are designed to incentivize activities within the ZONES platform. $LSX tokens have a limited total supply, ensuring scarcity and value within the ecosystem. The use of $HAMP tokens within the ZONES ecosystem creates a transparent and secure system for rewarding users based on their activities and engagement. The exchange rate between $HAMP and $LSX is fixed at 1:1, ensuring a stable and predictable value for the tokens.

Users can cash out their $HAMP tokens directly for $LSX or $CORE, providing liquidity and incentivizing users to earn more tokens. Each Zone has its own decentralized treasury, which is funded by the SuperAdmin and used to distribute $HAMP tokens to users who perform desired activities.

Any unused tokens at the end of the year can be carried over to the following year or burned to reduce the overall supply.

To further illustrate the tokenomics, let's consider an example of a law firm using ZONES for their operations. The law firm can create a Zone within ZONES and set up a rewards program using $HAMP. The SuperAdmin can specify certain activities, such as meeting a certain number of billable hours, completing specific tasks within a certain time frame, or contributing to the firm's knowledge base. When an employee completes one of these activities, they earn $HAMP, which can then be exchanged for $LSX. The $LSX can be staked in the pool with $CORE to earn additional rewards.

At the end of the year, the law firm can distribute a portion of the rewards earned through $LSX and $HAMP to their employees as a bonus, with the remainder being reinvested in the staking pool for the following year. This creates a self-sustaining rewards program that incentivizes employee activity within the ZONES ecosystem and rewards long-term participation and investment.

By utilizing the features of ZONES, such as incentivizing activities and customizable rewards, the law firm has been able to encourage active participation and engagement among its employees. The tokenomics and model of $LSX and $HAMP within the ZONES ecosystem create a thriving environment that incentivizes user participation, fosters collaboration, and drives the adoption of blockchain technologies.

$HAMP REWARD DEFAULT VALUES

VALUABLE ACTION	POINTS PER ACTION	% PER POINT	ACTIONS TAKEN PER HAMP	POINTS PER 1 $HAMP	REWARDS EARNED
VA 1	1	1%	100	100	1 HAMP
VA 2	1	1%	100	100	1 HAMP
VA 3	1	1%	100	100	1 HAMP
VA 4	1	1%	100	100	1 HAMP
VA 5	1	1%	100	100	1 HAMP
VA 6	1	1%	100	100	1 HAMP
VA 7	1	1%	100	100	1 HAMP
VA 8	1	1%	100	100	1 HAMP
VA 9	1	1%	100	100	1 HAMP
VA 10	1	1%	100	100	1 HAMP
VA 11	5	5%	20	20	1 HAMP
VA 12	5	5%	20	20	1 HAMP
VA 13	5	5%	20	20	1 HAMP
VA 14	5	5%	20	20	1 HAMP
VA 15	5	5%	20	20	1 HAMP
VA 16	10	10%	10	10	1 HAMP
VA 17	10	10%	10	10	1 HAMP
VA 18	10	10%	10	10	1 HAMP
VA 19	15	15%	6.67	6.67	1 HAMP
VA 20	20	20%	5	5	1 HAMP
TOTALS	100 pts	100%	-	-	-

ACCESS CONTROL & A DECENTRALIZED TREASURY

The SuperAdmin controls the rewards and can stake $LSX in the Zone community pool. The tokenomics incentivize activities within ZONES, with rewards distributed among employees and reinvested in the staking pool for the following year. By customizing the protocol, the SuperAdmin can tailor the rewards program to the specific needs and goals of their organization. This can lead to increased motivation and engagement among employees, as well as improved productivity and efficiency. Additionally, the customization can help to incentivize specific behaviors or outcomes that are important to the organization. This can lead to a more focused and aligned workforce, as well as improved outcomes overall. Overall, the ability to customize the protocol provides a powerful tool for SuperAdmins to optimize their rewards program and achieve their business goals.

The SuperAdmin plays a crucial role in the $HAMP and $LSX ecosystems, as they are responsible for customizing the protocol for their specific industry/workflow. The SuperAdmin can create custom smart contracts, set up governance rules, and define the parameters of the tokenomics for their organization. One of the primary benefits for the SuperAdmin is that they can create a more efficient and streamlined workflow for their organization, as they have the flexibility to design the protocol to fit the specific needs of their business. This can help to reduce costs, increase productivity, and improve overall efficiency.

Additionally, the SuperAdmin has access to various use cases that can benefit their organization. For example, they can use $HAMP and $LSX tokens to facilitate secure and efficient transactions, create loyalty programs, and incentivize customer engagement. By leveraging these use cases, the SuperAdmin can improve customer retention and drive revenue growth for their organization.

When combined with LP staking, the SuperAdmin can create a "decentralized treasury" tool for their organization. This allows them to pool resources and funds from stakeholders and community members, which can be used to fund various projects and initiatives. The LP staking model provides a secure and decentralized way to manage these funds, ensuring that they are being used in a transparent and efficient manner. Overall, the SuperAdmin plays a critical role in the success of the $HAMP and $LSX ecosystems, as they can customize the protocol to fit the specific needs of their organization and leverage various use cases to drive growth and innovation.

STAKING

Staking is a fundamental feature within the ZONES ecosystem that empowers users to participate in the Coreum network's security and governance while earning rewards. By staking $LSX tokens, users actively contribute to the decentralization and overall robustness of the Coreum blockchain.

STAKING PROCESS

To participate in staking, users need to acquire $LSX tokens. Once in possession of $LSX, users can stake their tokens by locking them into the staking contract. This process is done through a user-friendly interface within the ZONES platform, allowing even non-technical users to easily participate in staking.

VALIDATOR SELECTION

Through staking, users have the option to either become a validator themselves or delegate their $LSX tokens to existing validators. Validators play a vital role in securing the Coreum network and validating transactions. By delegating their tokens to established validators, users can actively support the network while benefiting from the rewards earned by the validator.

REWARDS DISTRIBUTION

As a reward for participating in staking and supporting the network, users receive a portion of the transaction fees and newly minted $LSX tokens as an incentive. The distribution of rewards is proportionate to the amount of $LSX staked by the user. This mechanism aligns the interests of stakeholders, fostering a strong and vibrant ecosystem.

SECURE & EFFICIENT

Staking in the Coreum network ensures a secure and efficient consensus mechanism that relies on the collective power of stakeholders. The proof-of-stake (PoS) model utilized by Coreum reduces the energy consumption associated with traditional proof-of-work (PoW) systems while maintaining the security and integrity of the blockchain.

FLEXIBILITY & LIQUIDITY

ZONES users can seamlessly unstake their $LSX tokens at any time after the initial locked timeframe, providing flexibility and liquidity to their holdings. This means users can easily access their staked tokens whenever needed without any delay.

SUPPORTING DECENTRALIZATION

By participating in staking, users actively contribute to the decentralization of the Coreum network. A diverse and distributed group of validators enhances the network's security, making it more resilient against potential attacks.

DECENTRALIZATION

Decentralized systems, such as those built on blockchain technology, offer several benefits over centralized systems, including increased security, transparency, and immutability. However, decentralized systems can also present certain challenges, such as lower performance and scalability.

In the case of ZONES, the platform utilizes a hybrid approach that combines the best of both centralized and decentralized systems. The platform is built on Cosmos SDK, a modular blockchain development framework, which allows for the creation of custom blockchain networks and decentralized applications. The use of Cosmos SDK allows ZONES to leverage the security and immutability of a decentralized system while also maintaining the performance and scalability of a centralized system.

In terms of data storage, ZONES uses Space and Time, a decentralized SQL database, which allows for the storage of large amounts of data in a decentralized and secure manner. This allows ZONES to maintain the security and immutability of a decentralized system while also maintaining the performance and scalability of a centralized system.

Additionally, ZONES uses a variety of security protocols such as access control, encryption, and key management to ensure that sensitive data is protected.

The use of a hybrid approach in ZONES allows organizations to take advantage of the security and transparency of a decentralized system while also maintaining the performance and scalability of a centralized system. This approach is ideal for businesses that require a high level of security and data integrity but also need to maintain a high level of performance and scalability in order to meet the demands of their operations.

DECENTRALIZED SQL

Lunarspace's use of Space and Time decentralized SQL databases is a key component in providing a secure and reliable ERP/CRM platform. The decentralized nature of the database ensures that data is stored across multiple nodes, making it much more difficult for any single point of failure to occur. Additionally, the use of a SQL-based database allows for easy querying and manipulation of data, making it more user-friendly for businesses and organizations of all sizes.

One of the main benefits of using Space and Time is the increased security it provides. By distributing data across multiple nodes, it becomes much harder for hackers to access or manipulate data. Additionally, the use of smart contracts on the blockchain provides an extra layer of security, as all transactions and data manipulation must be approved by consensus among the nodes.

Another benefit of using Space and Time is the increased scalability it provides. As the number of users and transactions on the platform grows, the decentralized nature of the database allows it to easily expand and handle the increased load. This contrasts with traditional centralized databases, which can become overwhelmed as usage increases.

Lastly, Space and Time allows for increased transparency and immutability. All the data and transactions stored on the database can be easily audited and verified, making it easy to detect any suspicious or malicious activity. The data stored on the database also cannot be altered or tampered with, providing an additional level of security and trust.

In summary, the use of Space and Time decentralized SQL databases in ZONES provides a secure, reliable, and scalable solution for storing and manipulating data, while being transparent, immutable, and easy to audit. This allows ZONES to provide a comprehensive and secure ERP/CRM platform with added benefits of web3 integration.

SCALABILITY

Space and Time is a decentralized SQL database that allows for horizontal scaling and high performance, which can help improve ZONES' scalability. However, it is not the only aspect of scalability that needs to be considered. Other factors such as the architecture of the system, load balancing, and the use of caching can also play a role in improving scalability. Additionally, while Space and Time can handle the scalability of data storage, it's important to also consider the scalability of other parts of the system such as the network infrastructure and the application logic. To improve the scalability of ZONES, we will consider implementing a microservices architecture, using a load balancer, and implementing caching mechanisms.

SECURITY

ENCRYPTION

ZONES uses industry-standard encryption algorithms to encrypt user data both in transit and at rest. This ensures that sensitive data such as customer information, financial data, and personal details is protected from unauthorized access.

AUTHENTICATION & ACCESS CONTROL

ZONES implements a robust authentication and access control system to ensure that only authorized users can access the platform and sensitive data. This includes multi-factor authentication and role-based access controls to limit the actions that different users can perform.

VULNERABILITY MANAGEMENT

ZONES has a vulnerability management program in place to identify, assess and mitigate security risks. This includes regular security testing, penetration testing and regular security audits.

COMPLIANCE

ZONES is built to comply with industry-standard data protection regulations such as GDPR, HIPAA, SOC 2, ISO 27001, and more, to ensure that user data is handled in a compliant manner.

FIREWALL & INTRUSION DETECTION

ZONES uses a firewall and intrusion detection system to protect the platform from external threats such as hacking attempts and denial of service attacks.

NETWORK SECURITY

ZONES uses secure protocols such as HTTPS, SSH, and SFTP to protect communication over the network. It also uses VPNs to secure remote access to the platform.

DATA BACKUP & DISASTER RECOVERY

ZONES has a data backup and disaster recovery plan in place to ensure that user data is protected against data loss due to hardware failure, natural disasters, or other unforeseen events.

SMART CONTRACT SECURITY

ZONES use smart contract security best practices such as formal verification and formal testing to ensure that smart contracts on ZONES are secure.

SECURE KEY MANAGEMENT

ZONES uses secure key management techniques to ensure that private keys are only accessible to authorized parties and are not compromised in any way.

Lunarspace is considering implementing additional security measures such as multi-factor authentication, network segmentation, and data encryption both in-transit and at-rest. Additionally, ensuring secure key management and access control are in place to protect against unauthorized access to sensitive data and systems is crucial. We also have a robust incident response plan in place, in case of security breaches or other security incidents occur.

IMPLEMENTATION

DISCOVERY

During this phase, the Lunarspace team works closely with the client to understand their specific needs and requirements for their ZONES platform. This includes identifying any unique business processes or workflow that need to be incorporated into the platform.

PLANNING

The team develops a detailed project plan that outlines the scope, timelines, and resources required for the implementation including a detailed breakdown of the tasks that need to be completed, along with the dependencies and milestones for each task.

CONFIGURATION AND CUSTOMIZATION

This phase involves configuring the platform to meet the client's specific needs. This includes customizing the platform's user interface, data model, and workflows to align with the client's existing business processes. Additionally, any necessary integrations with other systems or platforms are to be implemented during this phase.

TESTING

Once the platform is configured and customized, it is thoroughly tested to ensure that it meets the client's requirements and that all functionality is working as expected. This includes functional testing, user acceptance testing, and performance testing.

DEPLOYMENT

After successful testing, ZONES is to be deployed to the client's production environment. This includes configuring any necessary infrastructure, such as servers, databases, and networking, as well as performing a final set of acceptance tests to confirm that the platform is working correctly in the production environment.

TRAINING & SUPPORT

Once the platform is deployed, the team provides training and support to the client's users and administrators to ensure they can effectively use and manage the platform. This can include on-site training, documentation, and ongoing support and maintenance.

CONTINUAL IMPROVEMENT

The implementation process doesn't end with deployment; the team continues to work closely with the client to ensure that the platform is meeting their ongoing needs and that any issues or problems are quickly identified and resolved. The team will also work with the client to identify opportunities for continual improvement and to implement updates and new features as necessary.

ONBOARDING

ONBOARDING A NEW "ZONE" CLIENT

1. Lunarspace conducts a brief discovery process with the node to understand their specific needs and requirements for using ZONES .
2. Lunarspace creates a new instance of ZONES for the node and sets up a SuperAdmin user account with a temporary password.
3. The SuperAdmin receives an email with instructions on how to access the ZONES platform, including a link to the login page and the temporary password.
4. The SuperAdmin logs into the platform and is prompted to change the password to a more secure one.
5. SuperAdmin sets the basic configurations of the node.
6. SuperAdmin can invite new users and assign them roles and permissions.
7. SuperAdmin can set up custom workflows and configure the system to meet the specific needs of their organization.
8. SuperAdmin can begin uploading and processing documents, analyzing data, and engaging with patients or other stakeholders through the communication and collaboration tools provided by ZONES .

SUPERADMIN CREATING AN ADMIN WITHIN THE DAPP

1. The SuperAdmin logs into ZONES using their unique credentials, such as their email and password, and is directed to the ZONES Dashboard.
2. The SuperAdmin navigates to the "Users" section of the Dashboard and selects the option to "Add User".
3. The SuperAdmin is prompted to enter the new admin's information, such as their name, email, and role.
4. The SuperAdmin assigns the new user the "Admin" role and can also set specific permissions and access levels for the new admin.
5. The SuperAdmin inputs the new admin's information and clicks "Save" to create the new admin user.
6. The new admin receives an email with a link to set their password and log into ZONES.
7. Once the new admin has set their password, they can log into ZONES using their email and password and will have access to the features and permissions assigned to them by the SuperAdmin.

IMPACT

Lunarspace aims to have a significant impact on both businesses and the blockchain industry by providing a comprehensive and secure ERP/CRM platform with added benefits of web3 integration. The ZONES platform, which includes the ZONES Launchpad, dPASS SBT, Keplr wallet integration, and other relevant technologies, allows businesses to streamline their operations and improve their bottom line.

By accelerating blockchain adoption, Lunarspace is helping to pave the way for mainstream enterprise-level blockchain usage. This can have a major impact on the industry, as it can lead to increased adoption and acceptance of blockchain technology, ultimately driving its growth and development.

By providing a robust and secure platform that allows businesses to take advantage of the benefits of web3 technologies, Lunarspace is helping to bridge the gap between the traditional business world and the blockchain industry, ultimately driving innovation and growth in both spaces.

The integration of $HAMP, a Human Activity Mining Protocol, into ZONES provides a unique incentive for users to actively engage with the platform and earn rewards for valuable activities. This, in turn, leads to increased adoption and engagement within the platform.

By building ZONES and $HAMP on Coreum, we can take advantage of the fast, low-cost, and secure transactions offered by this blockchain. This will allow businesses to easily integrate ZONES and $HAMP into their existing systems, while also taking advantage of Coreum's robust ecosystem and decentralized infrastructure.

ZONES and $HAMP provide a powerful solution for businesses looking to improve their operations, increase engagement and drive adoption of blockchain technology. We look forward to working with the Coreum community to bring this vision to life and revolutionize the way businesses operate.

REFERENCES

COSMOS HUB

https://cosmos.network/

https://ignite.com/cli

https://v1.cosmos.network/sdk

https://ibcprotocol.org/

https://v1.cosmos.network/

https://v1.cosmos.network/resources/whitepaper

https://docs.cosmos.network/master/modules/mint/01_concepts.html#

COREUM

https://docs.coreum.dev/

https://docs.coreum.dev/guides/wasm.html#set-up-the-environment

https://sologenic.medium.com/

https://sologenic.medium.com/coreum-a-modern-layer-1-blockchain-fast-and-smart-f6049274accd

KEPLR

https://wallet.keplr.app/#/dashboard

https://docs.keplr.app/

https://github.com/chainapsis/keplr-wallet

JUNO

https://www.junonetwork.io/

https://docs.junonetwork.io/juno/readme

https://wallet.keplr.app/#/juno/governance

https://medium.com/@JunoNetwork

https://messari.io/asset/juno-network/profile

SPACE AND TIME

https://docs.spaceandtime.io/docs/connect-to-space-and-time#connect-via-jdbc

https://www.spaceandtime.io/about

https://docs.spaceandtime.io/docs

https://medium.com/@SpaceandTimeDB/decentralizing-data-adb385dcfebf