essentia.one

INTEROPERATING AND DATA FOR THE NEW INTERNET

Whitepaper

Draft

Version 1.20

Last Edit: 16th January 2018

0.0 ABSTRACT

The new paradigm-shift driven by blockchain technologies enables everyone to access a higher level of decentralization and privacy further to giving them the ability to build up and enjoy a completely new ecosystem.

While blockchain-enabled ecosystems grow, stabilize and mature further, users currently lack essential services and solutions that protect their privacy, data, IDs or assets. Such solutions can unlock more complex, yet smooth, simple and powerful, levels of interaction through blockchain-based and decentralized technologies/ systems.

With Essentia we build and offer a complete modular framework at the user's fingertips. The framework empowers the users with the full control and ownership over their data, IDs, information, privacy and assets. The Essentia framework, which also works as a decentralized Operating System, can be used indifferently by humans, machines, IoT devices and more. It integrates decentralized and centralized resources in one single "place", while it also safely connects and relates the user/machine's anonymous, pseudo anonymous or public digital IDs with their multiverse of data which is encrypted and stored on decentralized resources or locally.

Starting from just the Seed, using the fundamental cryptographic capabilities of sign/ verify/ encrypt/ decrypt, the Essentia Framework is able to bootstrap and scale up from a minimal headless setup to a full-fledged and personalized platform or infrastructure. This enables the user, being it a human or a machine, to seamlessly interact with decentralized resources, systems or services as well as with traditional or centralized variants. Only the user has access to their personal data, IDs, setups, preferences, wallets and more, from anywhere on the planet. Empowering him to easily connect and use on-chain, off-chain and cross-chain services or resources in a secure, private and reliable environment.

Essentia is the multi-asset and multichain decentralization swiss-knife that everyone can use and own. Decentralize yourself!

0.1

THE ESSENTIA FRAMEWORK IN A NUTSHELL

Essentia is a modular framework that is bootstrapped from a Seed by deriving its extended keys in a hierarchical deterministic way and by associating to them their related "data" that are available encrypted on decentralized storage or on local copies/backups. Such data are associated to their corresponding labelled extended keys and are interpreted by the system, and by its components, as "registry entries," configurations and content to be used when loading their related modules that compose the framework accordingly.

The Seed represents the minimum "quantity" of information that is necessary in order to always be able to access and use the connected data, IDs, assets, information, and so on, and to bootstrap the framework or parts of it, by always expecting the same identical result. All the data and communications are encrypted by default.

By being the only one who has full control and possession over the Seed, the user (or machine) is the only one with full ownership, access and control over their connected IDs, data, assets, and more.

The Essentia Core at the base of the framework manages the cryptographic layers and executes the sign/ verify/ encrypt/ decrypt/ functions that are the base operands on which the whole framework is built. Data and their operations are managed in isolated memory-boxes that are separated from each other also in order that each extended key is responsible only for its associated module's data and so, it can only execute operations with its pertinent data and memory-box.

The framework can be used for just loading and running a standalone module or a more complex setup. One or multiple instances of it can be preconfigured, automated and orchestrated in custom constellations, infrastructures or setups. Each time a Seed is loaded, a secure and trustable decentralized environment is bootstrapped accordingly to the datasets bound with it.

The modules that compose the framework can be on/off-chain or hybrid and they are responsible for the backend, frontend and input/output components. Among others, specific modules are responsible for cross-chain operations as well as dedicated modules available for integrating third-party decentralized or trusted traditional resources. Combinations of the available modules are able to fit a wide range of implementations, use cases, architectures and existing decentralized or centralized systems.

The Essentia framework can be used indifferently by humans or machines (servers, IoT, etc.) who both have available all of its resources. It can be fully accessed and used both via its CLI (command line interface) or its UIs (user interfaces), independently by the host system in use.

From a user perspective, the framework works as a modular Operating System where the kernel is represented by the Core, which becomes unique thankfully to the Seed. The Essentia Decentralized OS has a console, and also a Home Panel, that allow access and use of all of its features and functions. It can be loaded on almost every existing host machine architecture and OS, while being also suitable to run from any existing browser and on low-resource hardware/software environments.

Essentia provides a "decentralized desktop/server" experience to the users/machines and allows them to dispose, at their fingertips, and from the single place of a trustable, secure and private environment where they can interact with the multitude of existing decentralized services and resources, as well as with the traditional resources, which are integrated into the framework.

Multiple decentralized digital IDs and their Sub-IDs can be used and managed, even concurrently, by a single Seed. The IDs can be anonymous, pseudo anonymous or related with real-world data, accordingly to whatever the user wants to do with them.

The IDs, Login, KeyRing, Dapp Centre, Wallet, Account-less Exchange, DEX, Decentralized Storage, Prediction Markets, Decentralized VPN, Social Dapp, Crypto Nodes, and so on, are just some of the "frontend" modules that can be activated inside the Essentia framework with a simple click. The user just needs his Seed in order to always be able to load and use their own custom Decentralized framework from anywhere and on any kind of host machine or system, even virtualized environments.

The Seed can be also stored and used on a dedicated thumb-size hardware device, the Essentia Keyware, which isolates it from software risks, external environment risks and from bad actors. So, we can optionally separate the Kernel (Seed) of the framework on the hardware layer, in order to get an even more secure setup. At the same time, the whole framework can be deployed on dedicated hardware devices, the Essentia Eggs, that can be expanded through the Essentia Sense peripherals to interact with real-world inputs/outputs and to also connect them on-chain and cross-chain. The Essentia decentralized Global Position System (dGPS) is just one of the multiple implementations that allow Essentia to provide on-chain, off-chain and cross-chain Oracles.

Essentia is the very first Decentralized Modular Framework suited for both humans or machines which enables them to fully dispose of their decentralized digital IDs, data and assets, even cross-chain, while granting full privacy and granular control over every and each process or aspect of the framework itself.

The Essentia framework is like a decentralized OS that can integrate on/off/cross-chain resources in a single place, while also giving the user/machine a trustable operational environment to interact with them.

Essentia is the very first project that takes care of making decentralization concretely fungible, and effectively usable, by the users. For example, the user can load their multitude of preferred Dapps and resources and interact with them and all their respective accounts, configurations, preferences, and so on, by just accessing their Seed that loads the user's personal, custom and powerful Essentia Home Panel.

Because of its peculiarities and implementation, the Essentia framework cannot be compared to any of other existing projects. Essentia is able to integrate the other projects when necessary and, at the same time, it provides its services in a radical, decentralized way without, for example, asking to the user to rely on intermediary applications for doing simple tasks such as logins. These can instead be done in a truly decentralized way. By signing and verifying a message after a successful handshake/challenge between the parties.

Essentia is based on simple but powerful and easily replicable cryptographic operations: sign/ verify/ encrypt/ decrypt/. This allows everybody to always be able to access and use their "belongings" with minimum effort and in any future scenario.

At Essentia, we are moving back the weighing needle to the real sense of Decentralization. Where others are trying to re-centralize the decentralized resources by also creating artificial, if not imposed, needs of an intermediary to the users, we are moving back from this spiral-shaped paradox and we are rebuilding the Decentralization fungibility to a more radical and original layer. Exactly as it was intended to be since the beginning of this new dRevolution!

The Essentia Framework main properties are:

- Highly customizable Modular Decentralized Framework for humans and machines.
- Decentralized digital IDs for humans and machines.
- Trust-less, Trustable and Secure. Backed by strong cryptography.
- Private: only the user has full and total ownership and control.
- A single personal, custom environment for managing and interacting with multiple IDs, data, assets, decentralized and traditional resources, cross-chain resources and communications, and more.
- Decentralized, always available, un-stoppable.
- Cross-Chain, Multi-Chain and Multi-Protocol.
- Modular, Flexible and Scalable.
- Privacy-Focused, Anti-Censorship, Persistent, Not-Corruptible.
- Anonymous and encrypted by default.
- Agent Agnostic, Head/less and Automatable/Orchestrable, CLI, UIs.
- Open-Source and Open-Hardware, Essentia has both Software and Hardware implementations.
- Language Agnostic, Host Agnostic.
- Fault-Proof and Future-Proof.

INDEX

1.0 INTRODUCTION

Today's social and technological landscape is a fertile ground for new ideas, services and tools which are being developed quickly and continuously.

Users are presented with more choice than ever before. However, choice is provided at the expense of privacy, security and personal data usage. Often users and their data are considered the product to benefit from. Users' journeys are reliant on trusting third parties and intermediaries with the security of their data and software environments. In this context, being able to effectively protect users' privacy and data resembles a paradox.

Blockchain-based technologies have the power to solve the issues presented by this context. Yet in their current state, these technologies will not be sufficient nor effective until users are able to easily and proficiently interact with them. Decentralized solutions and intermediary-free services or tools are of critical importance to enable users to regain control over their freedom, privacy and what constitutes their digital "life pieces" including IDs and data.

1.1 CURRENT RISKS

Users' data including habits, personal information, locations, social relations, financial details represent the highest intrinsic value for the service providers and ill-intentioned actors. In this context, users are currently exposed to a multitude of risks which are directly and indirectly arising from their interactions with current IT solutions. Additionally, the centralization of resources and services combined with insecure data management practices pose several threats whereby the risk of having single points of failure with critical effects is highly enhanced.

Today users are affected directly or indirectly by service interruptions, data breaches and leaks, identity and money theft, scams, surveillance, content-censorship and privacy violations independently from their location and circumstances. Users are often and unknowingly giving away some, if not all of their rights regarding their identity, privacy, info and data. The user can feel powerless to protect and control their privacy and data.

1.2 CURRENT DEFICIENCIES

Every piece of the user's digital life is totally exposed and vulnerable. Users don't have full control or ownership over their own "digitalized life and belongings." They are forced to use and rely on the potentially dangerous, non-trustable and insecure existing solutions.

The lack of user friendly secure tools and services prevents users from effectively protecting themselves, and their data, while interacting with third-parties. Users are not given disclosure of how their data is being used or who it is shared with. Further, they can't trust the environments in which their connected information is managed and stored. They don't have any other choice but to always expose their data if they want to use or interact with the wide majority of the existing services.

Even if blockchain-based and decentralized technologies can lead the way out of this disturbing reality, they haven't yet reached a mature stage. Blockchain technologies lack layers of effective solutions enabling a positive, prolific and friendly interaction for users. To be finally freed, users require intermediate layers that connect them directly, easily and securely with decentralized resources, tools and services, as well as blockchain technologies.

1.3 WHY ESSENTIA

Essentia provides a complete set of solutions to users, decentralized service providers and their technologies. Essentia is the multi asset swiss-army knife solution that users always have in their pocket. Providing a gateway to decentralized resources and, through them, traditional internet services, while also having exclusive, full control, over their identity, digital life and data.

Essentia fills the solution gap of the current landscape and makes available a complete and versatile multi-chain framework. Its primary goal is to empower users with a toolset of decentralized solutions which seamlessly integrate with on-chain or off-chain systems, services or resources.

Essentia protects users' privacy, needs and security by developing a bleeding-edge and user-friendly framework. The Essentia framework gives full and granular control to users and enables them to act anonymously, pseudo-anonymously or publicly.

Users can decide whether or not to share their data in part or in full and when, how and where that data is shared. Beyond the ID system and its connected data and metadata tools or services, Essentia also provides secure, trustable and decentralized software environment solutions that can run on all major OSs and hardware.

Essentia is the connective tissue between the user, decentralized resources and traditional systems. It enables anyone, anywhere to take advantage in full of current and future technologies. The Essentia framework is the missing link that finally enables a trustable, simple and powerful way to interact with decentralized systems and everything that can be connected with them, whether software or hardware.

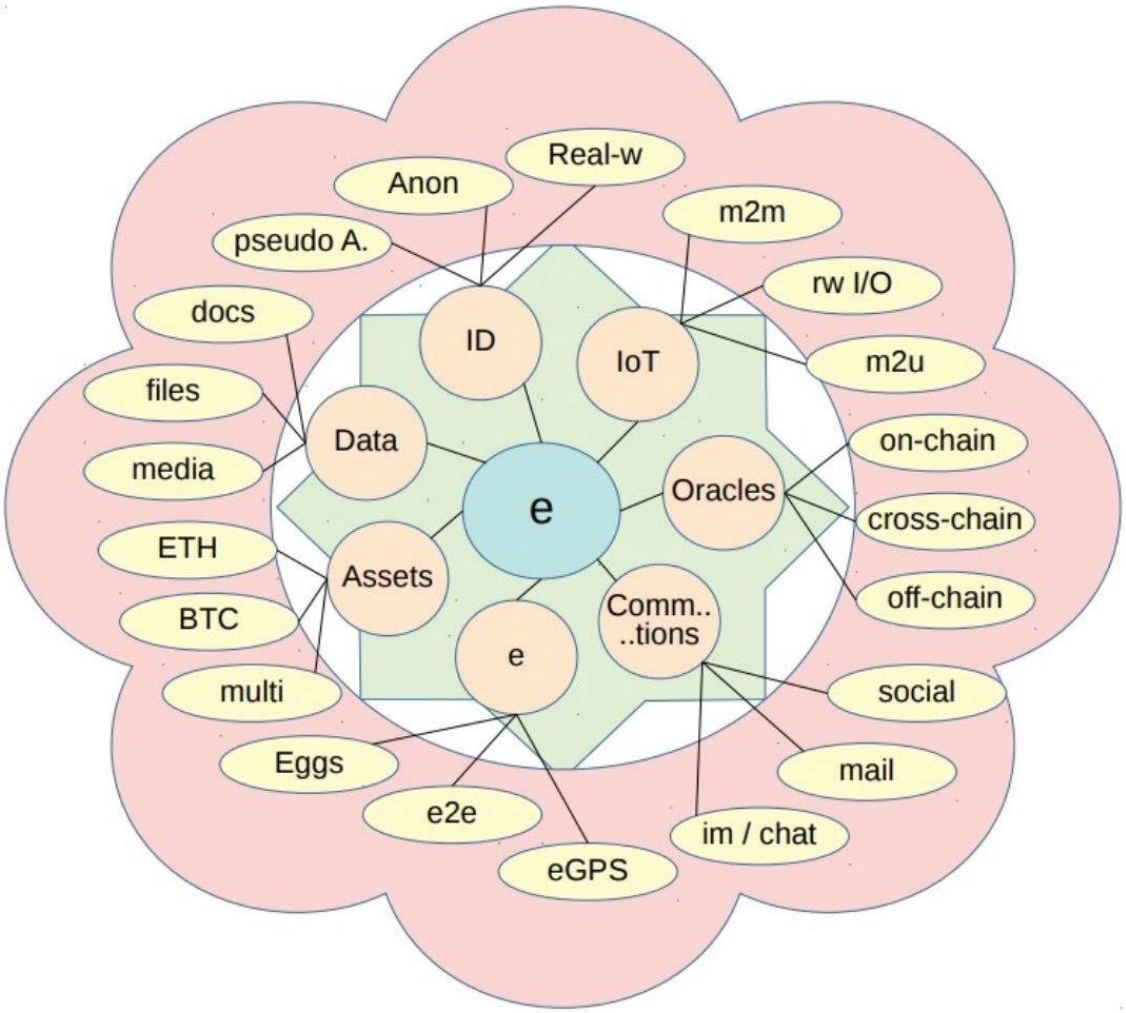
1.4 USER BENEFITS

With the Essentia framework, the users' IDs, data, desktop, files, accounts, software, wallets, logins and so forth, are not managed by third-parties or centralized services. They can simply follow the user and always be fully accessible and used from anywhere, in a trustable operating data environment.

Protected by cryptography and blockchain-based systems, the user is able to instantly and securely gain access to on-chain and off-chain third-party services and to interact with them. In each instance, deciding whether or not to share information or data.

The user's data can be encrypted and maintained on decentralized storage, removing any concerns regarding the persistence and content of the user's data. Users' encrypted data is not publicly connected with their IDs. Data can only be cryptographically accessed and used by its owner and by whom the owner decides to give access and partial or full control to.

Users can not only rely upon, and access, decentralized or virtualized resources and distributed computing but they are also able to bootstrap and use their own personal system. They can dispose of their data, and digital belongings, with the ease of use of a consumer solution and the full power that a decentralized framework can provide.



2.0 ESSENTIA DESIGN

Essentia is built with a modular architecture that can deploy, scale and configure its infrastructure to fit custom, personal setups or resource-intensive applications, automated or IoT frameworks. It fits the everyday use of beginner as well as expert users.

The Essentia framework can be bootstrapped or deployed in a browser, mobile platforms and Operating Systems, on any desktop, server or dedicated machine. It can be virtualized while eventually taking advantage of distributed computing. While evolving towards being a fully decentralized framework, Essentia also implements hybrid solutions that integrate smart contracts and other on-chain components. They work as bridges, and modules, executing the on/off-chain events and operations.

The ESS-ID, the core of the system, can be used and controlled via a dedicated hardware-key (wallet), offering a higher level of security to all the operations and interactions of the user. Essentia's components and all user data are maintained on decentralized storage and encrypted by default. The user is also able to back them up fully encrypted on local storage for offline access or use. This local data is synchronized with the data on decentralized storage at the first connection opportunity.

2.1 ESSENTIA'S FRAMEWORK OVERVIEW

The user - a human or a machine - can generate their own root ESS-ID (and then, eventually, the connected Sub-IDs) and be able to abstract, link, register, save their anonymous-identity, and eventually data, on the blockchain and on decentralized storage.

The ESS-ID has some unique properties and it is in full, exclusive ownership and control of the user. It is like a master-crypto-keyring that solely holds full control over the private-keys which manage the user's public identities and their operations, including data. The seed that fully recovers and restores the ESS-ID can be exported as a mnemonic list of words and allows the user to always and easily regain control over their IDs and data. The seed can be backed up on paper and using other secure methods.

It can be deployed and restored on the ESS-ID, on the ESS-Keyware as well as on other compatible software and hardware at a later stage.

The user is now able to use their decentralized ESS-ID to access the connected services and/or to link, access and use their data. The data is encrypted and stored on decentralized storage where it becomes persistent, censorship-resistant and always available. The user can restore and recall, access and use, all or some of their data from any physical location and using any kind of hardware or enabled system. Data can include wallets, documents, configurations, files and more. This is made possible through the ESS-Home and ESS-OS and their compatible/integrated alternatives.

From the ESS-Home in the enabled systems or, better, on the ESS-OS, the user is able to privately, safely and securely manage his data and identities as well as their connected software environments and configurations. The ESS-Home and the ESS-OS create a low-level (operating-system-level) trustable-data-environment which is integrated with blockchain-based technologies and applications, while also being enriched by many possible software modules and configurations.

The Essentia Home and Operating System are trustable, secure and user/dev-friendly operational and data environments that can be highly customized by the user. The user can use ESS-Home/OS for purposes such as trading, developing, managing wallets, payments and funds, contributing to the network, maintaining privacy, managing Sub-IDs, user-data, authorizations or accounts, logins and so forth.

The data of the ESS-Home/OS is available on decentralized storage and repositories, which can be downloaded and deployed from anywhere. The user can also use their own personal operating data-environment setups in live-versions on USB devices. These can run on most machines as well as be installed on dedicated systems and devices.

"Native" dedicated hardware for the Essentia OS are the ESS-Eggs. Open-hardware devices that can further extend the possibilities of the ESS-Home/OS and of its underlying blockchain technologies. The ESS-Eggs can also connect with the real-world through sensors and peripherals. They can become "autonomous" and have some additional interesting features that will be fully explored in the dedicated section.

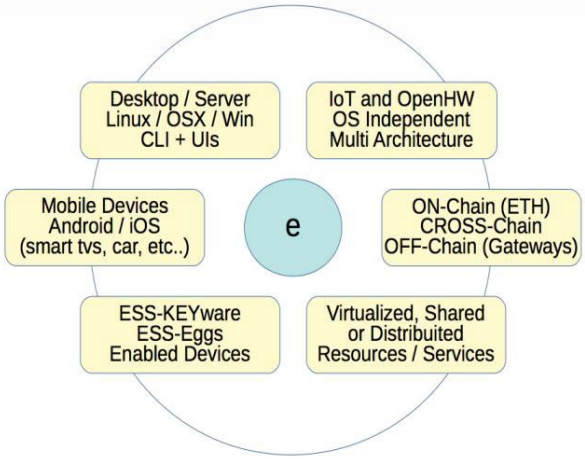
The ESS-Home and the ESS-OS also represent a powerful interface between the user, the blockchain world and its ecosystem. They are the native data environment where the blockchain applications and software express themselves and provide a seamless way to connect/interact with the real-world and with the users/machines too.

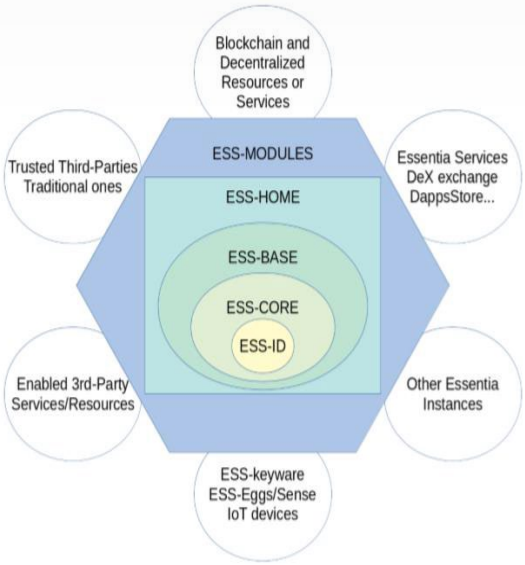
Through their modules, the ESS-Home and ESS-OS are able to integrate specific, trusted, third-party services and Dapps. These are in addition to the already integrated Essentia Dapp-Store, the DEX (Essentia Decentralized Exchange) and the modules governing IDs, Authorizations (Auths), wallets or the user-space system. Users can browse and interact with Dapps, use and manage their wallets, and interconnect with every available service on the blockchain.

The ESS-ID can be physically stored, used and managed within a hardware device, the ESS-Keyware. The Essentia Keyware is a dedicated mini device that acts as a secure keyring and can safely execute operations with the ESS-ID private-keys without revealing or exposing them in any way to external environments, systems or bad actors.

Through the ESS-Key, users can access/use/manage their IDs and any connected data. They can also use it as a "universal-key" to access/use/interact with third-party services such as Dapps and smart-contracts, extending to any on-chain or off-chain integrated resource, service or third-party.

Root-IDs can create Sub-IDs, managing the authorization layer of these Sub-IDs and their possible actions.





2.2

ESSENTIA'S FUNDAMENTAL LOGIC

The ESS-ID is an HD (Hierarchical Deterministic) "wallet" where the first sets of derived addresses that are generated from the Seed (the Extended Keys), correspond, and are linked at low-level, with the user ID(s), their data, setups and configurations, while also with the Essentia framework specific infrastructure, configurations and data.

From the Seed, and by following pre-determined derivation patterns, Essentia is able to bootstrap its framework, completely or partially, by using the generated keys for building up and deploying both the software operational environment and the user-space, that is populated with its data, content and configurations. Parts of the derived extended keys are used for defining the architecture and the logic of the framework. They can be assimilated to the system's registry entries/keys containing the operating system and application data that are then deterministically labelled, interpreted and understood by the system in order to bootstrap or deploy the framework and its components/modules.

The ESS-CORE module, that has the control over the ESS-ID, can execute sign, verify, encrypt and decrypt operations with the addresses (extended keys) belonging to the Root-ID (the Seed). Such s/v/e/d/ operations and their combinations are the lowest-level instructions and operands of the Essentia framework's brain.

The Root-ID (the user/machine who has full control and ownership over it) is the unique entity that can effectively dispose of the s/v/e/d/ layer of the keys, while every other entity is able to publicly verify its signatures.

The data connected to the Root-ID is stored encrypted on decentralized storage and becomes accessible through the ESS-SmartContracts, ESS-Bridges or ESS-Nodes. The data can be retrieved for bootstrapping the whole system or just parts of it, while the user can access and use just some of its features or services, depending on their needs.

The user only requires their ESS-ID since everything else, including data, IDs, Auths, wallets, logins, setups and configurations, can be fetched and/or deployed from the decentralized storage privately, securely and at any time.

When the first handshake between the Root-ID and the Essentia's blockchain interfaces takes place, the low-level data of the user and of their personal framework setup (contained in the Imprinting-Data packet) are fetched/downloaded, fully encrypted, from decentralized storage and are assigned to their respective logical parts (the keys/addresses associated with the infrastructure) for being interpreted and used. The components, the modules, their configurations and content are initiated and populated according to the user data and preferences. All this takes place in the background without bothering the user. This process is initiated by the Imprinting-Data which is the default bootstrap configuration that the user optionally sets up among their datasets, instead of the "empty" default one.

The decryption of the data happens locally in a protected software environment, as do the s/v/e/d/ operations connected with the keys. The Essentia framework, manages only encrypted data that is meaningless to everyone but the user/machine to whom they belong. The only entity that is able to decrypt them.

When we refer to the ESS-ID and the user's identity or identities, we are referring to anonymous IDs that are not linked in any way with real-world-IDs. Unless explicitly and deliberately linked by the user or via the optional ESS-Meta module, which can add additional layers of data to the IDs/Sub-IDs. This allows specific use of (encrypted) information for Know Your Customer and Anti Money Laundering requirements, as examples, when and where they are necessary and explicitly requested by the user. Please refer to the Technical Overview document for more detailed information on these topics.

In summary, a Root-Key and sets of its extended keys are derived from a Hierarchical Deterministic Seed, by following specific standard rules that grant and allow you to expect the same identical results at any access/use of the same Seed.

The Seed, in all of its forms, is the minimum piece of information needed to bootstrap the whole framework, or just part of it, and to access to all of its connected data.

After accessing to the Seed, one of the extended keys of the first derived set is used to retrieve, by a signature/verify handshake/challenge operation with a bridge, its Imprinting-Data packet which represents the first, minimal set of data that the system needs to further bootstrap the framework.

This minimal set of data includes the low-level configuration data such as the labels to be deterministically associated with the derived extended keys. By decrypting the Imprinting-Data packet, which is fetched through a bridge from the decentralized storage, if not locally available, the derived extended keys are labelled and associated with their respective data and pointers to data (links on decentralized resources). The labelled keys and their related content can be then understood and used by the system to bootstrap and configure the components and modules accordingly with the user/machine specific preferences, content and setups. So, the system uses some of the extended keys as its own registry-entries and associates to them their "values" that consist of data (local configurations) and pointers to additional data/resources that are available on the decentralized storage for later access/use.

At their bootstrap/launch, the Modules are injected with their respective configurations, data and content which are allocated (or linked) in the "memory-boxes" associated with their labelled keys (example: The content of the key labelled "Home-configuration" is passed to the Home module for initialization). By reading their configuration data and the pointers contained in their dedicated registry-entries/memory-boxes, the Modules also know "the routes" (pointers) to get access to the additional user/machine resources/data.

Sign/Verify/Encrypt/Decrypt/ are the low-level minimal operands that allow the whole framework to execute its operations. Data is available encrypted locally or in the decentralized storage and is locally decrypted in a secure environment, only when accessed or used.

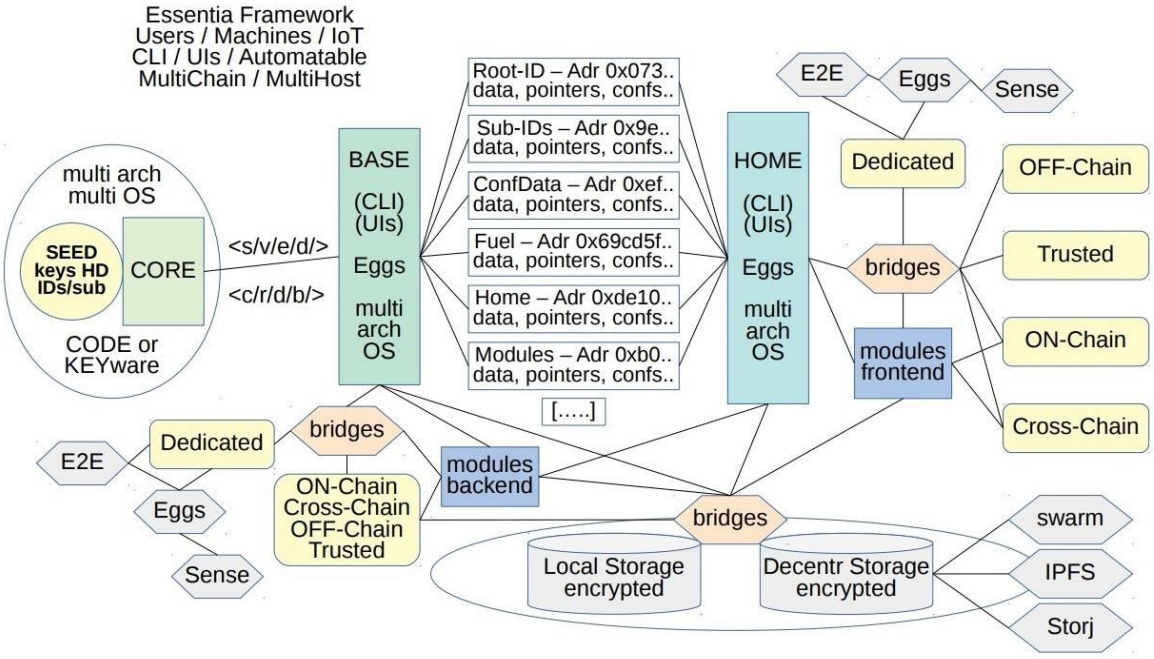
Some modules can also work as standalone Dapps/apps as they can be detached from the framework and used to perform dedicated or specific operations/interactions. For example, the ESS-Login module (eLogin) is available also as a standalone module which can serve as dedicated login/account keyring management, in both its software and hardware implementations.

The Essentia Framework is built on Ethereum and it follows, implements and strictly adheres to its protocol, its consensus rules and to the relevant involved Ethereum Improvement Proposals (or Bitcoin Improvement Proposals), and also to the other blockchain and industry specific protocols, standards and tools including Swarm, ENS and Whisper. The Essentia Seeds, Keys, Addresses, Formats and so on, fully comply and integrate with the Ethereum Protocol and they are perfectly and seamlessly compatible with it and with its implementations or connected services, tools and resources. This doesn't prevent, nor affect in any way, the cross-chain and multi-chain operability and fungibility of the Essentia Framework.

The aim of the Essentia Project is to develop, implement and share new protocol-level solutions for the Ethereum ecosystem and to propose, develop and contribute to new and/or existing EIPs, such as those already inspired by Essentia research and development, the EIP #725 and #735.

We are delighted to see the Ethereum Community and Developers already participating with the new topics that represent Essentia's focus: Identity, Data, Communication, Operational Environment, Privacy, Security and Decentralization.

2.3 MAIN COMPONENTS OVERVIEW: ESS-ID, ESS-HOME AND ESS-OS.



2.3.1 THE ESS-ID

The ESS-ID is the Identity core of the Essentia Framework. It works similarly to a HD wallet and can be created, or managed, via software or dedicated hardware devices. It has optional multi-signature features and capabilities.

At the logical level, it consists of a Root-ID that is a HD cryptographic master-key able to create and manage its Sub-IDs (public-keys) and their connected levels of operations and authorizations. Root-IDs can also create and manage individual and groups of Sub-IDs, including IDs from other users as well as their Auths. Multiple users are able to join groups or not and can eventually agree on specific Auths or operations by multi-signing the agreement itself and so adhering to its rules.

The base operations of the ESS-ID are to create/destroy/restore/backup/ the root-ID, to sign/verify/encrypt/decrypt/ data and to generate and manage its Sub-IDs with the same exact properties. An ESS-ID can be owned and controlled by a human user, a bot, machine, smart contract or IoT device and so on, depending on the setup and on the specific application of the Framework itself. The user can share Auths on IDs/Sub-IDs, create groups, set/enforce context-dependant rules, user/group depending rules and much more.

The interactions of the ESS-IDs between themselves and with the blockchain, or third-party services, are initially managed by hybrid on/off/cross-chain solutions, the ESS-Bridges/Nodes. They are then translated into on-chain autonomous smart-contracts that will just use the bridges to connect with traditional resources and systems when necessary or requested.

The ESS-ID is an HD Seed that can also be exported as human-readable string of words and can be backed up on whatever kind of support. The Root-Key is encrypted with a user-passphrase, that is a mandatory input for using the Root-Key itself for any operation, and with an additional encryption step called PIN. When the ESS-ID is on the ESS-Keyware, the passphrase can be a user input on a pad with an additional physical button for common and routine operations to be authorized.

As a dedicated hardware device, the ESS-Keyware, is able to operate with the ESS-ID in a native and secure way by isolating, at the hardware level, the sensitive/critical data of the keys from being exposed to the external context, environment or to bad actors.

2.3.2 ESS-HOME AND ESS-OS

The ESS-Home and the ESS-OS are the decentralized Essentia Data Environment and Operating System. While also being a blockchain gateway and interface, they offer a trustable and secure operating environment for the user and a native context for the Modules of the Essentia Platform/Framework to be deployed and operated.

The ESS-Home can be a standalone "Application" that runs inside other active operating systems (e.g. on mobile) or in-browser, while the ESS-OS is "The" operating system itself which can run in live-mode or installed on the vast majority of machines and architectures available. It can also be installed on dedicated open-hardware machines, the ESS-Eggs, which will be fully detailed in a following section. The ESS-Home and ESS-OS are available through decentralized storage and repositories and they can interact with the other ESS Modules also during a pre-boot/pre-deployment stage, in order to auto-configure themselves and/or to download additional data, based on the user's setup, for example.

The ESS-Home component is the main hub of the Essentia Framework where the ESS-Modules connect with each-other and the user's interaction can happen in full.

The ESS-OS and ESS-Home, as the complete framework, are open source and can be seamlessly configured in several flavours to be used by different user-types for their preferred activities or setups. The Home and OS are "clean" and user friendly, yet very robust, secure and trustable. This is while being a very powerful data environment, toolset and framework for the expert users and developers who can use, access and enable the more advanced configurations, depending on their needs.

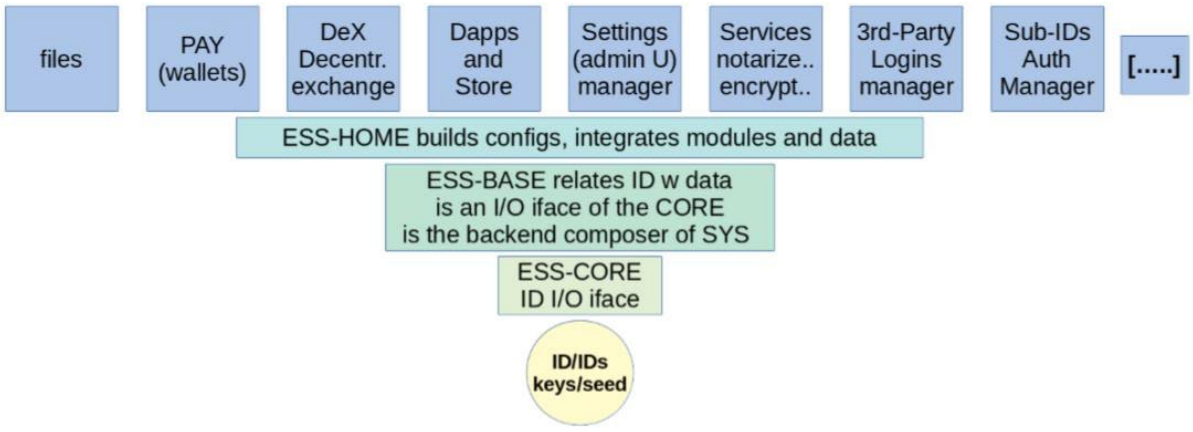
The ESS-CORE implements the low-level data environment which can be expanded with the ESS-Modules to meet specific needs or to deploy specific frameworks/operational environments. Until reaching a complete Home setup or a complete Operating System setup. The ESS-Home and the ESS-OS can then run on server/text (Command Line Interface) or graphical (Graphical User Interface/Browser User Interface) mode and can be accessed and used locally with Human Interface Devices (HIDs) or through the browser/console from other machines either on the local network or remotely, via internet, even in virtualized environments.

The user's configuration, and eventually their data, are encrypted and stored on decentralized storage through Modules such as ESS-Config and ESS-Data. The data can be kept in sync on different installations which the user can run concurrently on different machines. They can even be run concurrently on multiple instances on different machines and architectures. The setups can eventually share the configurations or the data and can be kept in-sync or automated/orchestrated through the dedicated Modules.

The ESS-Home/OS results in a powerful and trustable modular framework that can be configured/pre-configured to be used as a safe environment for executing transactions, trading, developing, blockchain browsing, dapping, mining, ID-control, smart contract interaction/deployment/test, education, demonstrations, anticensorship operations, for supporting decentralized blockchain technologies, or experimenting and developing with them, and much more.

The Essentia Home and Operating System are complemented and expanded by their Modules that, beyond connecting them with the ESS-ID and the rest of the Essentia Framework, allow trusted (blockchain-based or not) external services to be integrated, accessed and used. Some modules come in hardware form. The ESS-Sense are hardware modules that can extend the ESS-Eggs through their physical input/output layer by enabling the ESS-Platform, as well as the connected blockchain-based applications/smart-contracts, to interact with the real-world through sensor, motors, circuits, solar panels, network hardware and so forth.

At the same time, the ESS-Sense Modules can enable IoT devices to interact with each-other and to use the physical/hardware layer beyond the logical/software one already enabled by the ESS-Home/OS. The ESS-Sense modules will be fully described and detailed in their section.



2.4

MODULES OVERVIEW

Essentia Modules are the fundamental parts constituting the Essentia Framework constellation. They expand the ESS-ID and ESS-Home/OS functionalities by allowing them to interact with each-other and with decentralized resources or traditional ones.

Each module is competent for its functions and has its own properties inside the Essentia Framework. They can depend directly on one or both of the ESS-ID and the ESS-Home/OS. Additional modules, like the hardware ones, will be detailed in the following sections.

The main modules that compose the Essentia Framework are:

1

ESS-CORE

The software or hardware-implemented core-module that allows the ESS-ID to perform its fundamental and basic operations. The ESS-ID destroy / restore / backup the Root-ID (Seed), is able to sign / verify / encrypt / decrypt data and to generate / destroy / manage the Sub-IDs which have the same exact properties (s/v/e/d/). It also has multi-signature creation / management capabilities.

2

ESS-BASE

The module (strictly) connected with the ESS-CORE that effectively functions to create/manage/restore/use/ the Root-ID by dialoguing with the CORE. It is written in several languages/libraries so it can fit browser in-memory modules, dedicated hardware, apps, Dapps or plugins in order to adapt to existing frameworks and operating systems. It doesn't store user data or keys but just allows the user to operate at the top level with the Root-ID and Sub-IDs keys that are loaded and managed on the ESS-CORE. ESS-BASE is like an interface to the ESS-CORE API. Only the CORE touches the keys and the seeds. BASE is able to bootstrap the rest of the system by starting with the ESS-Home module and its imprinting data.

3

ESS-HOME

The main module that connects ESS-BASE and the CORE functions to all of the other modules and components along with the top-level user interaction. It also hosts the modules that connect/integrate with the trusted third-parties. Please take a look at the preceding section dedicated to the ESS-Home description and details.

4

ESS-Auth/Groups

The module that manages the creation/editing/deletion of the Authorizations connected with the IDs and of the controlled or participated (multisign) groups. This module also governs the creation, modification and deletion of groups and their lists of users, Auths and hierarchical statuses.

5

ESS-PAY

The module to manage access to encrypted crypto-wallets which can be created, accessed or managed by the Root-ID only. It can be extended with plugins to connect and manage other payment processors and/or wallets. It can also be connected with hardware devices such as Point of Sale solutions through its plugins.

6

ESS-FUEL

The module that takes care of the ESS Token layer inside the system. It manages the token transactions and flows within the modules/components, outside of the local system and for the users. It is also the antispam layer and the resource-balancing of the Essentia framework. Please refer to the Token Overview section for further details.

7

ESS-META

The module that connects IDs or Sub-IDs to additional metadata layers is also used to interact with resources or external third-party services. The metadata added to IDs help their use in real-life applications and/or in automated interactions. The user or the service provider can apply additional/different/custom rules based on metadata including reputation and good-payer. AML/KYC layers can be enabled with the ESS-META module also.

8

ESS-Talk/Docs/Stamp

The messaging (pgp-style) module to sign / encrypt / decrypt / verify the data/docs/messages exchanged with other IDs and third-parties. It communicates with other IDs, reads data, notarizes data, and encrypts data for specific IDs or groups, as well as integrating with existing on/off-chain services like Whisper.

9

ESS-PingPong

The module that carries simple data-communication protocols with limited data/bandwidth. Like Twitter for IoT devices but also for streamed inputs/outputs and link-like/real-time protocols.

10

ESS-DNS

Essentia Decentralized Name System, the module that translates and makes discoverable/reachable IDs' public keys (the user's addresses or accounts) into top-level/human-readable web domains or as blockchain resources. It can connect with Ethereum Name Service (ENS) and/or bridge with other services or resources.

11

ESS-BRIDGES(s)

The bridges are off-chain, hybrid or on-chain modules that connect both decentralized resources and the traditional centralized resources with Essentia. The Bridges also enable and allow cross-chain and multi-chain operations with and between different blockchains and protocols. They also make decentralized resources available in traditional environments and vice versa (e.g. gateways). While enabling, for example, a Dapp or a smart contract to be reached from the "outside" in order to also interact with the traditional resources or input/output systems.

12

ESS-DATA

The module that manages the core data infrastructure. It connects the ESS-IDs with its decentralized data, manages the public access and I/O to the decentralized public data and storage. It interconnects decentralized resources (smart contract, oracles, IDs, IoT, etc) with data, data-environments and between each other. Data can be encrypted or not, depending on the user or service. Access to data is granted through its plugins, that can manage the data inputs/outputs, track data on decentralized resources such as Swarm, IPFS, and Torrents while also be connected with traditional resources. The ESS-DATA module also has several other plugins that allow it to interact with the other modules of the Essentia Framework.

13

ESS-CONFIG

The module that handles all the configuration data. Including data used to store configuration-files such as software environments configs / variables, customizations of the data pointers and also of the build-scripts or other internal system's configuration. ESS-Config also takes care of configuration files and preferences of third-party services and resources.

14

ESS-AUTOMATE

The module that takes care of the deployment, automation or orchestration of standalone, or constellation, instances of the framework. It takes care of all the operations that involve automated or machine setups as well as infrastructure, IoT or complex and distributed setups.

16

ESS-ORACLE(s)

The module(s) that act as oracle(s) inside the Essentia Framework but also inside wider blockchain contexts such as cross-chain environments. ESS-Oracles can be used and accessed by third-parties and they can be connected to sensors and traditional resources.

18

ESS-DEL

The module that handles the deletion/reset of data such as all-configs, all-media, all-Sub-IDs and also of the delete-everything function and platform-reset. With time the module will also have more granularities in controls.

20

ESS-CLOCK

The module for taking into account and serving the blockchain-relative time. It serves as the reference heart-clock in a decentralized environment. As an example, it can count the time starting from the block's numbers as the reference measure-unit. It serves the BLOCKTIME that is used as the main time reference for timestamping data / events / connections.

15

ESS-TOOLS

The modules that are specifically meant for enabling smooth connection and interaction with the Essentia Framework to develop on it or to use it (or just modules/parts) from "the outside".

17

ESS-SYNC

The module that manages specifically the synchronisation of data between modules / OS / multi-devices with ESS-Home / OS, external services and others.

19

ESS-INIT

The module that takes care of the re/initialization of configurations, data, files and others. It also has or connects to default data-models for the framework / components / modules initializations and for setting up the default configs and infrastructure bootstrap parameters.

21

ESS-PROC

The module that checks the state and the sanity of the other modules. It can eventually re-initialise them or, until further developments of the Platform, it serves as system-monitoring and health-check of the system in each of its parts.

22

ESS-KEYRING

The module for storing / managing / accessing / use the additional key pairs (controlled and owned by the Root-ID directly or through its extended keys) used for s/v/e/d/ specific data/files e.g. large / routine. ESS-Keyring data is stored encrypted in decentralized storage and can be only accessed / decrypted / used by its owner.

23

ESS-dGPS

The module that takes care of connecting and making available on the decentralized resources, on/cross-chain, the dataflows coming from the ESS-dGPS Nodes. It not only manages the interconnection and transport layer of the data but also controls the quality and reliability of them, among the other functions.

24

ESS-OS-Config

The module that connects with ESS-Config and deploys the user-desired configurations to the ESS-OS. It connects with ESS-Sync, ESS-DATA and ESS-REPO to also download data or to keep the ESS-OS updated and in sync with other systems.

25

ESS-OS-DATA

The module for data not managed/included in the ESS-DATA module. It can manage additional data including static content, backups and media-files which are then used for the Essentia Operating System layer itself.

26

ESS-REPO

The module that tracks updates and manages the repositories of the ESS-OS and of other components and modules of the Essentia Platform.

27

ESS-CRYPTO

The module that enables blockchain(s) - based specific code/software to run through its sub-modules. It can enable full-node operations, mining-node operations, mining-pool-node ops, simplified payment verification (spv) wallet-node operations, smart-contract specific or API specific operations and more.

28

ESS-MAKE/SMITH

The OS-Maker module that makes build-scripts/images/ISOs/etc of the ESS-OS or ESS-Modules and that eventually configures them accordingly to the user setup by connecting to the ESS-ID, ESS-Config, ESS-DATA and the other relevant necessary modules. It can also make regular builds that are then uploaded on the decentralized storage as well as prepare the images to deploy on a live/installation USB-hdd, and so on.

29

ESS-NET

The module that enables specific network layers for the ESS-Home/OS and that can be expanded by its sub-modules to connect and use the tor-network or Virtual Private Networks (VPNs), as examples, through its plugins.

The Essentia Framework modules can have plugins that extend their functionalities and allow the whole ESS-Constellation to self-maintain to remain active and fully operational. The specific module plugins and extensions are detailed in the Technical Overview document.

2.5 ADDITIONAL COMPONENTS OVERVIEW: ESS-KEYWARE, ESS-EGGS AND ESS-SENSE

2.5.1 ESS-KEYWARE

The ESS-Keyware is a thumb-size device that allows full control and interaction with the ESS-ID and represents the most secure and safe way to use the ESS-ID as it offers all the advantages that a separate and isolated hardware device can offer to a critical/sensitive system like the Essentia Platform.

The ESS-Keyware is a dedicated hardware device able to operate with the ESS-ID in a native and secure way by isolating, at the hardware level, the sensitive/critical data of the seed and of the ID-keys. This prevents them from being leaked or exposed, to the external environment or to bad actors. The ESS-Keyware enables the ESS-ID to have the exact same features that the ESS-CORE software module (or smart contract) offers but implemented on a specific and dedicated "keyring-device". It allows the user to fully own, control and interact with the ESS-ID by providing the ability to create/destroy/restore/backup the Root-ID, to sign/verify/encrypt/decrypt data and to generate, use and manage its Sub-IDs with the same (s/v/e/d/) properties.

By managing its ESS-ID, the ESS-Keyware device enables the user to interact with the ESS-Platform services, with external/third-party or traditional services or any connected services. It can simply be used to login somewhere, or it can be used to fully deploy a system, or to move funds, execute payments, unlock a car and start its engine and for many other applications both in real-life use-cases and in automated/IT ones. If lost or destroyed, the ESS-Keyware internal content and keyring can be easily and simply restored/recreated starting from a backup-seed of the ESS-ID.

2.5.2 ESS-EGGS

An ESS-Egg is an open-hardware machine that is specifically crafted and used for running the ESS-OS and its hardware/software modules. The ESS-Eggs are affordable but very powerful devices that can be highly customized and expanded. They are power-efficient and low-power consuming while also being able to have high-frequency multi-core CPUs that can perform like a desktop machine. They are low-weight and can be easily transported by users.

Specific ESS-Eggs are designed for light operations or for being integrated with IoT devices. These Light-Eggs consist of "lighter" hardware or specific hardware-modules that can only run, manage or connect specific ESS-Modules (or their adaptations) to low-resources environments.

The ESS-Eggs are the native Essentia and Blockchain(s) hardware devices while they can also represent native Blockchain IoT hardware solutions and layer. Thanks to them, the user can always deploy a trustable environment that perfectly fits their needs, allowing him to rely on a dedicated hardware layer that executes the more critical framework operations in its secure context.

The Eggs can be used to host mission-critical operations or for more common uses, if not to deploy custom infrastructures or architectures. They can easily and efficiently scale up existing infrastructure setups and services. Private or public custom solutions and services can be hosted on them, from a production environment to a development or authoritative one. The Eggs provide all the features of the Essentia Framework but also on a dedicated hardware layer so they can also be automated, orchestrated and controlled and via the dedicated modules.

Real-world applications can use or integrate the ESS-Eggs in order to connect to decentralized resources, to deploy their physical "blockchainized" IT infrastructure, to integrate their existing services with decentralized capabilities, and so on. For example, a decentralized renting company can integrate a specifically configured Egg to their door-locks in order to connect them to the smart contracts that manage the logical layers of the transactions between the user and the rented resource.

An ESS-Egg is just as flexible as the Essentia Framework in its possible configurations and setups. Just a single module, let's say the ESS-Crypto module that runs a full Ethereum node, can be loaded and executed on an Egg. Custom constellations and infrastructures of automatable, orchestrated Eggs can be deployed in order to meet a wide range of the user's needs.

The ESS-Eggs represent another layer where the ESS-Oracles reside. An Egg can be configured to act as an Oracle by loading the pertinent modules and resources. The Eggs can be expanded with their hardware modules, the ESS-Sense, and when appropriate configurations and setups are applied, the Oracles can also manage real-world input/output dataflows.

2.5.3 ESS-SENSE

ESS-Sense are the Essentia Framework hardware modules that include peripherals and external devices able to expand the logical/software level of the Essentia Operating System, or of the ESS-Eggs devices, to the inputs/outputs of the external real-physical-world.

They also enable machine-to-machine connection and communications while adding some fundamental features to the IoT devices and infrastructure ecosystem already empowered by the ESS-OS and/or ESS-Eggs.

The Essentia Framework main properties are:

- HID modules | screens, touch, keyboards etc.
- Mining modules | can be external units too
- Energy modules | solar-panel units, battery packs, etc.
- Ambient sensors modules | temp, pressure, light, humidity etc.
- Network modules | mips hw, lan (ev), bees, antennas, etc.
- Movement modules | gyroscope, accelerometer, compass etc.
- Location modules | gps, antennas etc.
- Shell and fan/dissipater modules
- miscellaneous modules

The ESS-Sense enables on-chain and cross-chain Oracles to become a reality by connecting real-world inputs/outputs to the blockchain and decentralized resources. ESS-Nodes that are equipped with Sense peripherals can link and inject their sensors' dataflows directly into smart contracts or they can enable module/API access to them. The identity, trust and stability of the Oracles are ensured, and enforced, via smart contracts that also take care of the availability and quality of the data from the Nodes/Oracles.

As an example; an infrastructure of Egg-Nodes equipped with a Sense GPS antenna and module can be used as an Oracle providing position data on the Ethereum blockchain. Beyond other uses, the accuracy and the quality of the data from the antennas are controlled via smart contracts that make sure that the reported position of the nodes remains (for example) within 5 meters of tolerance.

Such an infrastructure and its Oracles can be used to have anonymous absolute and relative position data on-chain and cross-chain. So, for example, a smart contract can determine anonymously if a user has effectively reached a position (e.g. triangulating his location on a rented apartment) without worrying about alternate or counterfeit data. Position-triggered events can be trustfully processed in its code.

Welcome to the Essentia Decentralized Global Position System (dGPS)

2.6 ESS TOKEN OVERVIEW

Disclaimer:

This is just an overview for the sake of continuity in this document. Please refer to the dedicated token documents for the complete details and information about the ESS Token and its Economics

ESS Tokens are ERC20-based Ethereum Tokens. Each and every ESS Token is equal to each other and has exactly the same qualities, properties, and characteristics. After the ICO period ends, no additional ESS Tokens will be created.

ESS Tokens, managed mainly by the ESS-Fuel module, work like the fuel of the Essentia Framework by accomplishing several functions. The Token layer and its functions will be gradually introduced and implemented throughout the development of the entire Essentia Platform.

The ESS Token is devoted to the following main tasks (these will be enabled step by step, and made operational, as the underlying infrastructure is ready for them):

- As an antispam measure which prevents the abuse of the resources, or services, of the Essentia Framework (or anything connected to it). It also protects Essentia from several kinds of attacks that can be perpetrated against the infrastructure, the users or the connected resources and parties.
- It works as the fuel that keeps the Essentia Platform/Framework alive and provides the ability to access, maintain or use some specific component, feature or service.
- It is used to pay for extra internal resources (e.g. for more storage than the minimum included for free or for more configuration slots than the free default).
- It is used to pay for renting additional decentralized third-party resources connected to Essentia (e.g. computational power or running instances of the components).
- It is used to pay for generic but trusted third-party services connected to Essentia.
- It is used to pay for extra goods/services/content/fees on the in house ESS-Dapp-Centre and also on the integrated, and trusted, third-party marketplaces and Dapp-Stores. It can also be used for the integrated DEX, the Essentia Decentralized Exchange.
- When someone offers dedicated or additional resources to the network, they can be compensated with ESS Tokens by the users. When the project will reach a more mature stage, the contribution to the network through offering additional resources will be incentivized. The ESS-X Tokens will be minable and will represent the reward for contributing to the network by running a node, offering instances or virtualized resources, providing calculation power, securing the network, running specific decentralized services, or tools, and other activities to yet be defined. This will work as the incentivization system for resources specifically dedicated and connected to the Framework.

- ESS Tokens that are used at system level (like for antispam mechanisms for example), as well as the ESS Tokens collected as fees, will be periodically redistributed to the ESS Token owners in equal proportion to their stakes.
- This enables a distributed governance layer that can involve users into project and platform related decisions. Users are able to make proposals, or vote for them, with the ESS Token.
- They are able to have a significant role in the evolution of the ecosystem they actively use and appreciate.
- The token allows the user with a minimum stake of coins, beyond participating to the governance of the project, to have access to dedicated exclusive resources.
- It is used to manage the user's reputation layer by discouraging bad actors that would fear, not only eventually losing their reputation, but also a corresponding part of their tokens stake.
- Part of the ESS Tokens allocated to the Essentia Project will serve as a reserve dedicated to foster and promote the adoption of the Essentia Framework and to reward the active users for their involvement, participation or resource-sharing.

For further and more detailed information about the ESS Token and the Token Sale, please refer to their respective dedicated documents.

3.0 ESSENTIA SYSTEM PROPERTIES

Besides the indirect properties that the Essentia Platform has as a whole, there are some direct, and inherently connected, system properties and peculiarities which we are going to discuss individually in this section of the document.

3.1 TRUST-LESS, TRUSTABLE AND SECURE

Essentia is trust-less. It does not require the user's trust at all because it doesn't know nor manage any of the user data/information in the open. Everything, included the code, is stored and available encrypted on decentralized storage or resources. The user only has to trust him-self and eventually the system and the environment which hosts its operations. The Essentia Framework is an agnostic system that just cares about everything going well and working perfectly smoothly.

Essentia is Trustable since it provides the user with full transparency over every process, operation and component. It also offers fine-grained, complete and transparent control over what is happening at every level.

Essentia is Secure because its code is audited, community reviewed and built on solid cryptographical basis as well as on top-tier blockchain technologies, solutions and resources. It implements and applies the EIPs, BIPs as well as other industry standards, recommendations, best-practices and protocols.

3.2 PRIVATE: ONLY THE USER HAS FULL OWNERSHIP AND CONTROL

The user, being - it a human or a machine - is the only one who, by having exclusive access to their Seed, has full control and ownership of the data, privacy, content, wallets, identities, accounts or any other connected resource.

By being the sole entity who can cryptographically execute valid sign, verify, encrypt and decrypt operations with its ESS-ID (the Seed), the user is free to dispose of his digital belongings the way they prefer. A user is free to decide "how" to specifically interact with particular third-party resources or services.

3.3 DECENTRALIZED, ALWAYS AVAILABLE

The Essentia Platform and its resources are based and available on/through blockchain technologies. The data and the code are available through decentralized storage and the user is able to access them from anywhere as well as being able at the same time to store and backup them physically where preferred.

The user, by owning and controlling their own ESS-ID (the Seed), doesn't need to be concerned with, or interact with, intermediaries and third-parties when accessing, or disposing, of their digital belongings.

The user can finally, be fully decentralized!

3.4 CROSS-CHAIN, MULTI-CHAIN AND MULTI-PROTOCOL

The Essentia Framework can operate on multiple blockchains and with multiple protocols. It is designed to enable, and allow, cross-chain operations and transaction while also being able to integrate resources from different blockchains under the same secure environment. For example, ESS-Oracles become cross-chain resources too.

Essentia Nodes and Bridges are responsible of the interoperability between different protocols, together with the other backend modules and components of the Essentia Framework. The user, or machine, can easily and seamlessly operate on different blockchains. They can select and control which services (modules) to enable in their custom setup of the framework, or which resources to integrate.

3.5 FLEXIBLE AND SCALABLE: MODULAR

By design, the Essentia Framework is modular and highly customizable. This allows it to be a fit for the maximum variety of use-cases and to smoothly scale in order to adapt even to the most exotic scenarios.

The high-availability of the Essentia resources, which are always accessible from anywhere, together with their unique design as replicable and resources-sharing modules, allows the whole framework to be used concurrently on (possibly) infinite instances. At the same time, the Essentia Framework and its components are able to run on most virtual, physical and operational environments, including distributed.

3.6 PRIVACY-FOCUSED, ANTI-CENSORSHIP, PERSISTENT, NOT-CORRUPTIBLE

From its own internal point of view, Essentia manages and connects only meaningless, encrypted data. The focus on the user's privacy is of fundamental importance and it is absolute on the Essentia Platform, at every level. The default settings and operations are made using strong encryption and they only happen in protected memory, or files, if not directly on dedicated hardware devices.

The content or data, once on the blockchain, cannot be censored, erased or modified. The data becomes persistent in the decentralized cloud. Where only the user can access and control the data, platform components and smart contracts.

The Essentia Framework implements a Privacy-by-Design approach on all of its layers. The Framework doesn't manage any data, metadata or information which can be related to the user/machine. The framework doesn't need to do this for any reason, at any point of its development and life.

Essentia is fully compliant with all the European Union Regulations and Directives on Privacy and Data Protection. Essentia is strictly compliant with the EU Regulation 2016/679 too.

3.7 ANONYMOUS AND ENCRYPTED BY DEFAULT

The Essentia IDs are anonymous by design as they are agnostic black-boxes that don't care about real-world identities and data relations. Only the user can decide whether or not to eventually associate the ID(s) with their real ones or with sets of data/metadata. Therefore, multiple IDs can be managed, even concurrently, and provide granular control over their "state" of being anonymous, pseudo anonymous or reflecting real-world data.

The Essentia Platform "simply" takes care that everything, from the components to the modules, to the connected resources, is working like planned and it is secured by strong encryption, cryptography and by the available blockchain technologies.

All data and communications are encrypted by default.

3.8 AGENT AGNOSTIC, HEADLESS AND AUTOMATABLE / ORCHESTRABLE

The Essentia Framework is designed and developed to be used in several modes, by both humans and machines. It has both the primary, and low-level, CLI (command line interface, headless) mode and also the UI (user interface) modes which can be conveniently, and independently, used depending on the setups or needs. The Framework just doesn't care about "whom" is using it.

Every component/module of the Framework can be accessed, used and automated through the CLI mode. The IDs layers are Agent Agnostic just like the whole framework; they can be used indifferently and in their fullness by IoT agents, human users, server setups and so on.

The Seed can be configured, or preconfigured, to meet the custom setups or configurations that will be then deployed on the host system(s).

Standalone instances, or infrastructure setups, of the Essentia Framework can be fully automated and orchestrated, starting from their initial deployment on the host(s). Master/Slave and Chef/Puppet logics are applied to the Essentia Framework. While build-scripts and receipts can also be embedded in the Seed's data sets, to be applied/deployed at bootstrap.

3.9 OPEN-SOURCE AND OPEN-HARDWARE / SOFTWARE AND HARDWARE

The Essentia code, components and modules are open-source. Everyone is allowed to check and review the code, to adapt it for their personal purposes, to improve it or to do whatever they like with it.

At the same time, the hardware components of the Essentia Platform are based on open-hardware machines that offer the same guarantees of the open-source but at the hardware level.

All of this is just part of the total transparency the user absolutely deserves and that Essentia delivers.

3.10 LANGUAGE AGNOSTIC, HOST AGNOSTIC

The Essentia Framework consists of a set of protocols, sub-protocols and standards which make use of low-level cryptographic operands together with other open source code, protocols and standards to operate and build up the framework with its functions and features.

The framework modules and components can be implemented and written in almost any programming language that is compliant and adheres to the set of "conventions" determined by the Essentia consensus. Almost any language is suitable for implementing the framework and for developing on it. This makes of Essentia a multi-language or language agnostic platform.

Even at the hardware level (architecture) the Essentia Framework is compatible with almost every existing solution available today. The framework can run on most host architectures and operating systems. Being able to scale and be fully functional by adapting to low-resources setups, as well as to virtualized or distributed high-end setups. This trait makes it Host-Agnostic.

3.11 FAULT-PROOF

The Essentia Framework is built and based on open source code, standards and implementations. Moreover, Essentia is built on the minimal set of cryptographic operations of sign/verify/encrypt/decrypt that are executed with (and by) the seed and its derived extended keys.

These properties allow each and every user to access, recover, extract or use their data, IDs, assets, and so on, whenever they like and by using any kind of tool, implementation or custom code. So whatever scenario could appear in the future, there will not be any data loss and a block of code will be sufficient to have full access to everything connected to the Seed.

3.12 FUTURE-PROOF

Essentia is built on, consists of and develops bleeding-edge solutions and technologies which aim to be, not only one step ahead of the current environment, but also to be disruptive and revolutionary. Essentia puts its efforts, and commitment, into further developing decentralized ecosystems while offering future-proof solutions which keep the user at the centre of their vision.

4.0 APPLICATIONS AND USE-CASES

Among the several possible applications and use-cases of the Essentia Framework or its components, we picked-up the most significant of them in the following sparse list. This list expands the cases cited in the above sections of this document. The goal is to offer concrete starting examples, and also inspiration, for making use of the Essentia technologies and/or to start developing on them.

- The decentralized user becomes a reality!
- The user can access, from a single place, the multitude of integrated and connected decentralized Dapps, services, resources, and so on, by having everything at their fingertips.
- A CLI (command line interface) and UI (user interface) experience are finally available to access the decentralized ecosystems from a single decentralized framework that works like an operating system.
- Multiple IDs and their Sub-IDs, being them anonymous, pseudo anonymous or linked to real-world data, can be created, used and managed from a single place. Where users can selectively use them, even concurrently, in order to connect and interact with the services/Dapps integrated on the Framework.
- Reliable and trustable Oracles, available on/off/cross-chain, become a reality. Oracles can also take advantage of connections with real-world inputs/outputs through the ESS-Sense components.
- The user can login to services - use and configure them - without revealing their private details and data, or they can use their favourite exchanges, payment services and wallets without concern.
- The user can connect to the Internet, even at boot, via decentralized or custom VPN services or via the TOR network.
- The user can register/access their accounts instantly and seamlessly on every integrated trust-less service. The external applications can enable Sub-IDs to be KYC/AML compliant by binding an additional level of data/metadata to a Sub-ID with the ESS-META module or by supplementing ESS-REP the reputation model, beyond the other possible implementations.
- The user can organize and manage IDs or sets of IDs that are devoted to, and setup for, specific use-cases. For example, users can setup an ID to connect to his bank accounts, one for web shops, one for games, and so on, by having each ID related to the additional data/metadata that they decide to share with the third-party services/resources.
- Cross-chain communications and transactions become possible and can take advantage of a transparent and reliable transport layer. Multi-protocol along with multi-chain communication and transaction solutions enable the sharing of resources between different decentralized systems and environments.

- A good can be rented or used without revealing data if no problems occur with the use of it or with its payment.

- Insurance and/or other external, third-party smart contracts can finally manage "decentralized-users".

- The user can access their ESS-ID and, for example, can retrieve their personal data, decrypt and deploy the data or simply access/use the data from anywhere. The business man, or developer, who travels, can securely restore and redeploy their systems without the need of carrying a device, or storage solutions for content. The user can travel with empty hardware and then recover their data on destination, decrypt and interact with the data, then reset the host system.

- The user can always utilize a trusted, secure, powerful, custom, ready-to-use and user-friendly framework.

- The user can manage "his things" through his ESS-ID from anywhere, securely. For example, the user can access home controls, assign company's assets (e.g. cars), move funds, do trading, Dapping, access/use decentralized and third-party services and deploy their setups.

- The user can create/share authorizations on IDs and Sub-IDs, create groups, set/enforce context-dependant rules or user/group-dependant rules.

- Constellations of machines, or IoT devices, of any complexity level can be automatically deployed, managed, setup, and orchestrated through ESS-IDs, the ESS-Components and the ESS-Modules.

- Absolute or Relative positions on the Earth can be retrieved by using the Essentia decentralized Global Position System (dGPS). A constellation of Nodes with a Sense GPS antenna that are controlled during their operativity by smart contracts, create an on/cross-chain network that works exactly like a decentralized position system. Data from the GPS Oracles can be used by smart contracts to interconnect with real-world inputs/outputs in a decentralized, trust-less way. If, for example, an Oracle misbehaves or changes its position, a smart contract would put it immediately offline. The non-corruptible position of a machine/user can be reliably used to trigger events connected with real-world assets (e.g. rented resources, locks, cars, etc).

- The user can be actively/passively involved with the network. They can participate or contribute to the Essentia Network and to other decentralized systems by; mining, running nodes, offering local or virtualized resources for the network, computational power, decentralized storage, and so on.

- The user is able to load and run a full Ethereum node, other blockchains nodes, or decentralized VPN nodes, even concurrently, by just activating the corresponding modules on the Essentia Framework.

- Existing services and implementations, even hardware examples, can be easily integrated into the framework and be interconnected with decentralized and cross-chain resources. ESS-Eggs can provide a perfect solution for this connectivity.

- A smart-door lock system of a decentralized renting company can be easily interfaced with Ethereum, or other blockchain, with the addition of an ESS-Egg properly configured for the scope.

- The user can book a flight without giving their personal data to the operator/website by being the assignee of just an anonymous ticket. The user data, forwarded (encrypted) to them by the aero company, can be only decrypted and read by the relevant authorities. The user's data can eventually be accessed (by the authorities) only in presence of an alert, depending on the local jurisdiction. User privacy is fully respected.

- The user orders a shipment and their data is "gradually" revealed to the intermediate actors during the freight trip. For example, the shop knows (is allowed to decrypt) just the user's country, the central postal-service knows also the city, the local post the street, and the postman knows the door-number and the initials of the recipient's name.

- Decentralized resources can be rented, paid for and accessed in a decentralized way!

- A company can remotely manage the access and the use of its assets, such as cars, by employees. The manager can enforce limits and rules based on context or he can assign the assets while controlling their authorizations, enabled-times, and so on.

- With an ESS-Keyware, or ESS-ID enabled-system, the user can use ATMs/POSs without the need of a card, without leaving data to the intermediaries along the way and without needing accounts on them.

- The user, with an ESS-Keyware or dealing with an ESS-ID enabled system, beyond being able to manage doors and car locks, they can also access, use or pay for third-party services and resources such as; car-sharing, energy-sharing, Wi-Fi sharing or any service enabled/integrated for such use-cases.

- Services like Airbnb, for example, can integrate the Essentia Platform, or just some parts of it, and they can allow ESS-ID users to directly access/use their services. The user can be selectively authorized by the smart-contracts that manage the locks, the payments and the insurances to allow access to the rented resources and to use them. On the completion of the contract, if no issues have occurred, the user's data are not revealed to any party.

- The preferences, options or configurations on third-party services, such as logins and accounts, are accessible by the user through the ESS-ID. The third-party will challenge the ESS-ID to sign/encrypt configuration files that it can decrypt/verify. The user is the only one that can cryptographically sign (and so authorize) for the configuration/preference changes. The data can be stored
encrypted and the third-party doesn't need to store user login data, configurations or other kind of data.

- The user can access and use the integrated Dapps Centre module, where they can browse Dapps or interact with them along with other decentralized resources or services.

- Users can have at hand their digital assets and they can securely, yet easily, dispose of them for trading or for other tasks including using them on the integrated DEX or account-less exchange.
- The user is able to sign, verify, decrypt, encrypt, or timestamp, a document (or any other data) and he is able to manage its authorizations such as who can access/read/use them and when.
- Decentralize Yourself!

5.0 ESS LEGAL AND CROWDSALE

5.1 GENERAL INFORMATIONS

The ESS does not have the legal qualification of a security, since it does not give any rights todividends or interests. The sale of ESS s is final and non-refundable. ESS s are not shares and do not give any right to participate to the general meeting of Essentia One. ESS cannot have a performance ora particular value outside the Essentia One sagl network. ESS shall therefore not be used or purchased for speculative or investment purposes. The purchaser of ESS is aware that national securities laws, which ensure that investors are sold investments that include all the proper disclosures and are subject to regulatory scrutiny for the investors' protection, are not applicable.

Anyone purchasing ESS expressly acknowledges and represents that she/he has carefully reviewed this white paper and fully understands the risks, costs and benefits associated with the purchase of ESS.

5.2 KNOWLEDGE REQUIRED

The purchaser of ESS undertakes that she/he understands and has significant experience of cryptocurrencies, blockchain systems and services, and that she/he fully understands the risks associated with the crowdsale as well as the mechanism related to the use of cryptocurrencies (incl. storage).

Essentia One sagl shall not be responsible for any loss of ESS or situations making it impossible toaccess ESSs, which may result from any actions or omissions of the user or any person undertaking toacquire ESSs, as well asin case of hacker attacks.

5.3 RISKS

Acquiring ESS and storing them involves various risks, in particular the risk that Essentia One may not be able to launch its operations and develop its blockchain and provide the services promised. Therefore, and prior to acquiring ESS s, any user should carefully consider the risks, costs and benefits of acquiring ESS in the context of the crowdsale and, if necessary, obtain any independent advice inthis regard. Any interested person who is not in the position to accept orto understand the risks associated with the activity (incl. the risks related to the non-development of the Essentia One platform) or any other risks as indicated in the Terms & Conditions of the crowdsale should not acquire ESSs.

5.4 IMPORTANT DISCLAIMER

This white paper shall not and cannot be considered asan invitation to enter into an investment. Itdoes not constitute or relate in any way nor should be considered asan offering of securities in any jurisdiction. The white paper does not include nor contain any information or indication that might beconsidered as a recommendation or that might be use to base any investment decision. This document does not constitute an offer oran invitation to sell shares, securities or rights belonging to Essentia One or any related or associated company. The ESS is just a utility token which can be used only onthe Essentia One platform and is not intended tobe used asan

The offering of ESS on a trading platform is done in order to access the Company's platform, purchase services related exclusively to the latter and not for speculative purposes. The offering of ESS on a trading platform is not changing the legal qualification of the token, which remains a simple means for the use of the Essentia One platform and is not a security.

Essentia One is not to be considered as advisor in any legal, tax or financial matters. Any information in the white paper is given for general information purpose only and Essentia One does not provide with any warranty asto the accuracy and completeness of this information. Given the lack of crypto-token qualifications in most countries, each buyer is strongly advised to carry out a legal and tax analysis concerning the purchase and ownership of Essentia Ones according to their nationality and place of residence.

Essentia One today is not a financial intermediary according to Swiss Law and isnot required toobtain any authorization for Anti-Money Laundering purpose. This qualification may change in case Essentia One will offers services which are tobe considered as qualifying a financial intermediation activity. In this case, the use of Essentia One services may require the positive conclusion ofanAML/KYC identification process.

ESSs confer no direct or indirect right to Essentia One's capital or income, nor does it confer any governance right within Essentia One; an ESS isnot proof of ownership or a right of control over Essentia One and does not grant the controlling individual any assetor share in Essentia One, orin the Essentia One network. An ESS does not grant any right to participate in control over Essentia One's management or decision-making set-up, or over the Essentia One network and governance to the Purchasers.

Regulatory authorities are carefully scrutinizing businesses and operations associated tocryptocurrencies in the world. In that respect, regulatory measures, investigations or actions may impact Essentia One's business and even limit or prevent it from developing its operations in the future. Any person undertaking to acquire ESS must be aware of the Essentia One business model, the white paper or terms and conditions may change or need tobe modified because of new regulatory and compliance requirements from any applicable laws in any jurisdictions. In such a case, purchasers and anyone undertaking to acquire ESS acknowledge and understand that neither Essentia One nor any of its affiliates shall be held liable for any direct or indirect loss or damage caused by such changes.

Essentia One will do its best to launch all of its operations. Any person undertaking to acquire ESS acknowledges and understands that the Company's platform will function and provide access and services asat the launching of the crowdsale. Depending on further potential developments of the Company's platform, other services may be released and offered to the Users.

On concluding the Commercial Operation, these tokens will be issued by a technical process referred toas a «Blockchain». This isan open source IT protocol over which the Company has no rights orliability in terms of its development and operation. The token distribution mechanism will becontrolled by a Smart Contract; this involves a computer program that can be executed on the Ethereum network oron a blockchain network that is compatible with Smart Contract programming language. They acknowledge and understand therefore that Essentia One (incl. its bodies and employees) assumes no liability or responsibility for any loss or damage that would result from orrelate to the incapacity to use ESS s, except in case of intentional misconduct or gross negligence.

ESS s is based on the Ethereum protocol. Therefore, any malfunction, unplanned function orunexpected operation of the Ethereum protocol may cause the Essentia One network or Essentia Ones to malfunction or operate in a way that is not expected. Ether, the native Ethereum Protocol account unit may itself lose value in a similar way to ESS s, and also in other ways.

5.5 REPRESENTATION AND WARRANTIES

By participating in the crowdsale, the purchaser agrees to the above and in particular, they represent and warrant that they:

- Have read carefully the terms and conditions attached to the white paper; agree to their full contents and accept tobe legally bound by them.
- Are authorized and have full power to purchase ESS according to the laws that apply in their jurisdiction of domicile.
- Are not a U.S. citizen, resident or entity (a"U.S. Person") nor are they purchasing Essentia One or signing on behalf of a U.S. Person.
- Are not a Chinese resident or entity nor are they purchasing the ESS or signing on behalf of a Chinese resident.
- Live in a jurisdiction which allows Essentia One to sell ESS through a crowdsale without requiring any local authorization and are in compliance with the local, state, and national laws and regulations when purchasing, selling and/or using Essentia Ones.
- Are familiar with all related regulations in the specific jurisdiction in which they are based and that purchasing cryptographic tokens in that jurisdiction is not prohibited, restricted or subjectto additional conditions of any kind.
- Will not use the crowdsale for any illegal activity, including but not limited to money laundering and the financing of terrorism.
- Have sufficient knowledge about the nature of the cryptographic tokens and have significant experience with, and functional understanding of, the usage and intricacies of dealing with cryptographic tokens and currencies and blockchain-based systems and services.
- Purchase ESS because they wish to have access to the Essentia One platform.
- Are not purchasing ESS for the purpose of speculative investment or usage.

5.6 GOVERNING LAW –ARBITRATION

The Client acknowledges and accepts that the Essentia One ICO operation is taking place within a Swiss legal environment that is still under development. The Parties agree to seek an amicable settlement prior to bringing any legal action. All disputes arising with the with papers provided, shall be resolved by arbitration in accordance with the Swiss Rules of International Arbitration of the Swiss Chambers of Commerce in force on the date when the Notice of Arbitration is submitted in accordance with these Rules. The arbitration panel shall consist of one arbitrator only. The seat of the arbitration shall be Chiasso, Switzerland. The arbitral proceedings shall be conducted in English.