

Florida International University
Knight Foundation School of Computing and Information

Sciences Software Engineering Focus

Final Deliverable

Project Title:

Nextcloud App: Exploring Integration with Kaseya VSA

Team Members: Alexander Chirinos, Julian Arias, Nicholas Samaroo

Product Owner: Masoud Sadjadi

Mentor: Masoud Sadjadi

Instructor: Masoud Sadjadi

The MIT License (MIT)
Copyright (c) 2016 *Florida International University*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and

associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Abstract

In the ever-evolving landscape of IT management, the seamless integration of tools and platforms becomes imperative for optimizing workflow efficiency. This abstract explores the integration of Kaseya VSA X and 9 with the Nextcloud server, presenting a robust solution for unified IT management and collaboration.

Kaseya VSA X stands as a comprehensive IT management platform, empowering organizations to efficiently manage and monitor their IT infrastructure. Meanwhile, *Nextcloud* has emerged as a leading open-source cloud collaboration platform, offering secure file sharing, communication, and collaboration features.

The integration process involves bridging the functionalities of Kaseya VSA X and 9 with Nextcloud to create a unified environment. By doing so, organizations can streamline their IT operations, enhance collaboration, and improve overall productivity. Key components of this integration include:

Single Sign-On (SSO): Achieving a seamless user experience by implementing single sign-on capabilities, allowing users to access both Kaseya VSA and Nextcloud with a single set of credentials.

Asset and Device Management: Leveraging Kaseya VSA's robust device management features to enhance Nextcloud's capabilities in tracking and managing assets across the organization, ensuring a unified view of the IT landscape.

Automation and Workflows: Integrating Kaseya's automation capabilities with Nextcloud

workflows to automate routine tasks, reducing manual intervention and improving overall efficiency.

Security Integration: Enhancing the security posture by integrating Kaseya VSA's security features with Nextcloud, ensuring a comprehensive approach to data protection and compliance.

Monitoring and Reporting: Utilizing Kaseya VSA's monitoring and reporting tools to provide real-time insights into the performance of both IT infrastructure and Nextcloud services, enabling proactive issue resolution

Introduction.....	4
User Stories.....	4
Project Plan.....	5
System Designing.....	6
System Validation.....	6
Glossary.....	7
Appendix A.....	8
Appendix B.....	9
Appendix C.....	14
Appendix D.....	16

Introduction

Current System

The current system is built on the Nextcloud server, featuring a robust integration with Kaseya VSA 10 API and VSA 9 API. This integration enhances the capabilities of the Nextcloud platform, providing advanced functionalities related to IT systems management.

Purpose of New System

Our system is designed with a primary focus on simplicity, offering users an intuitive and easy-to-navigate platform for accessing key information. The goal is to provide a clean and straightforward user interface (UI) within the Nextcloud server, coupled with seamless integration with Kaseya VSA 10 API and VSA 9 API.

USER STORIES

The following section provides the detailed user stories that were implemented in this iteration of the Nextcloud App: Exploring Integration with Kaseya VSA project. These user stories served as the basis for the implementation of the project's features. This section also shows the user stories that are to be considered for future development.

Implemented User Stories

User Story ID	Belongs To	Description	Assignee
#01	Developers	Review Pull Requests for each Sprint	All members
#02	Developers	Post a new issue to GitHub	All members
	Developers	Task #1: Welcome Spring 2023!	Entire team

#583	Developers	Task #2: Understanding Nextcloud and Kaseya	Entire team
#594	Developers	Create Agents	Entire team

#553	Developers	Understanding and gaining access to VSA X	Entire Team
#479	Developers	Started working on Front end for VSA 9	Alexander Chirinos
#602	Developers	Integrated API for VSA X	Nicholas Samaroo

PROJECT PLAN

This part explains how we planned and executed the project using agile development methods. It covers the detailed planning of our sprints. We also discuss the specific software and hardware components chosen for the project in this section

Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project:

- Nextcloud
- Kaseya REST API
- Github
- Javascript
- Vue
- Visual Studio Code

Sprints Plan

Sprint 1

Team was assigned to a new project in which we integrated Kaseya VSA API into our own application housed on the Nextcloud platform..

Sprint 2

User Story 1 – Familiarize yourself with the Nextcloud Platform!

Log into our individualized Nextcloud platforms, research and familiarize yourself with how to use Nextcloud .

User Story 2 – Familiarize yourself with Kaseya Rest API

After the meeting, we were instructed to do research on the Kaseya Rest API documentation . .

Sprint 3

User Story 1 – Deploy Agents on the Kaseya VSA 9 platform

Each individual should launch an agent on the Kaseya VSA 9 platform..

User Story 2 – Figure out the problem with deploying the agents to Kaseya VSA

Agents should be effectively launched on the Kaseya VSA platform by following sprint.. .

Sprint 4

User Story 1 – Continue studying the Kaseya VSA Rest Documentation

Research and learn how to effectively integrate Kaseya Rest API into our application .

User Story 2 – Build a simple frontend for our API integration

The UI for the application should be built using Vue.js to accommodate the Nextcloud Platform.The UI would have enough data to present a simple version of VSA.

User Story 3 – Learn Javascript for the Backend API integration.

The application would be in the form of a web app, therefore javascript was needed to effectively integrate the API.

Sprint 5

User Story 1 – Integrate the API for VSA X

Getting the essential user information and details to effectively integrate VSA X API into our application.

User Story 2 – Upgrade the frontend

Vue.js was utilized to upgrade the frontend adding key information and a better interface for the user

Sprint 6

User Story 1 – Fix bugs in the frontend.

The frontend was remodeled and cleared of most bugs

User Story 2 – Integrate our application into Nextcloud

The application was to be put onto the Nextcloud server.

Sprint 7

Work on completing the final documentation as well as the videos and our individual posters.

System Design

This segment provides insights into the design choices made during the development of this project. It delves into the outlined and explained architectural patterns, presenting the entire system through a package diagram and providing explanations for the subsystems. Lastly, the discussion revolves around the design patterns implemented in the project

Architectural Pattern

We adopted a Microservices Architecture for our project, leveraging the Kaseya API to retrieve the required data and display it on our frontend. This approach involves breaking down the application

into small, independent services that communicate through APIs. Our backend operates as a distinct service interacting with the Kaseya API, while the frontend consumes the data provided by this backend service. This architectural choice enhances scalability, flexibility, and maintainability in our system

System and Subsystem Decomposition

We utilize the Vue.js framework for the frontend and Node.js for the backend. The Kaseya API was employed to fetch the required data, with our backend serving as an independent service responsible for interacting with the Kaseya API. The frontend, built using Vue.js, seamlessly consumes the data provided by the backend service. This architectural approach enhances the scalability, flexibility, and maintainability of our system, aligning with the technology stack comprising Vue.js and Node.js

Design Patterns

For our project, we implemented a straightforward design featuring a left sidebar for easy navigation between various pages. Additionally, the login screen allows users to switch between different versions through a home menu. This design aligns with the principles of the Navigation Design Pattern, providing a user-friendly and intuitive way to move between different sections of the application

System Validation

In our project, GitHub served as our version control system for collaborative development, with each user story assigned a dedicated branch. We exclusively used Visual Studio IDE for debugging and testing, working with the Vue.js framework. Changes underwent verification before merging into the main development branch, a process requiring two team members' collaborative review. During reviews, the app was locally tested on computers using individual Docker containers containing a Nextcloud local server, ensuring task completion without introducing new issues. Bug reports were promptly addressed. Throughout the semester, the development branch seamlessly merged into the release branch on GitHub. Unlike traditional code testing, our team utilized Docker containers for file sharing among group members. Each team member's Docker container hosted a local instance of the Nextcloud server for testing. Additionally, we established an online Nextcloud instance for user interaction and testing. This approach facilitated effective collaboration, thorough testing, and the release of our work to a broader audience at the end of the semester, tailored for computer use with Visual Studio IDE

GLOSSARY

API (Application Programming Interface):

- Definition: API, an abbreviation for Application Programming Interface, encompasses a set of functions and procedures. These enable the development of applications that can access the features or data of an operating system, application, or other services.

Authentication:

- Definition: Authentication is the process of verifying the identity of a user to confirm whether the user is who they claim to be.

Cloud Functions:

- Definition: Cloud Functions provides a server-less execution environment for creating and linking cloud services. It allows developers to build and connect various cloud services without the need for server management.

Dependencies:

- Definition: Dependencies, in software engineering, refer to the reliance of one piece of software on another. It reflects the interconnected nature of software components.

Expo:

- Definition: Expo is a platform that facilitates the development and rapid iteration of applications based on React Native and JavaScript.

Feature:

- Definition: A feature in software represents a specific capability or functionality that the software offers to its users.

Framework:

- Definition: A framework serves as the fundamental structure underlying a system, concept, or textual representation. It provides a basis for building and organizing software.

IDE (Integrated Development Environment):

- Definition: An Integrated Development Environment (IDE) is a software application that offers comprehensive tools and facilities to computer programmers for efficient software development.

Product Owner:

- Definition: The Product Owner is the individual responsible for guiding changes and communicating with developers about modifications needed in software development.

UI (User Interface):

- Definition: UI, an abbreviation for User Interface, is the visual display through which users interact with a system, application, or device.

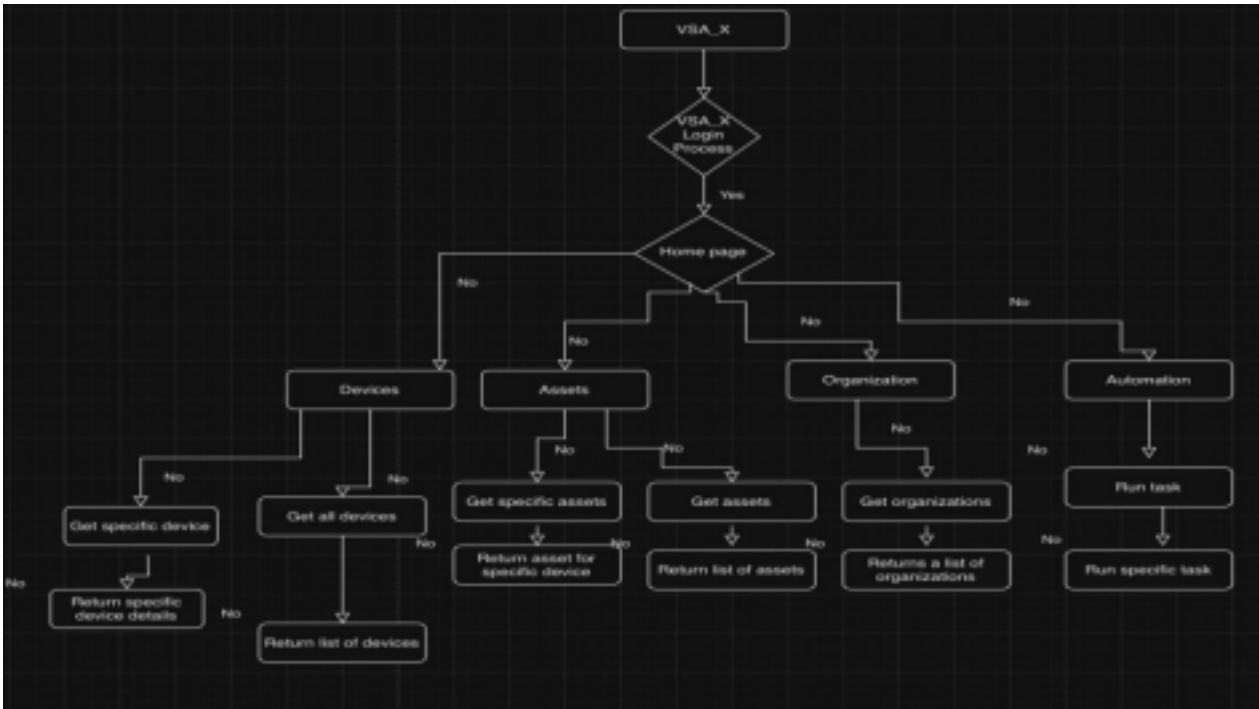
Diagram (Unified Modeling Language Diagram):

- Definition: A UML diagram is a visual representation based on the Unified Modeling Language. Its purpose is to depict a system, showcasing its primary actors, roles, actions, artifacts, or classes. UML diagrams enhance understanding, modification, maintenance, or documentation of system information.

APPENDIX

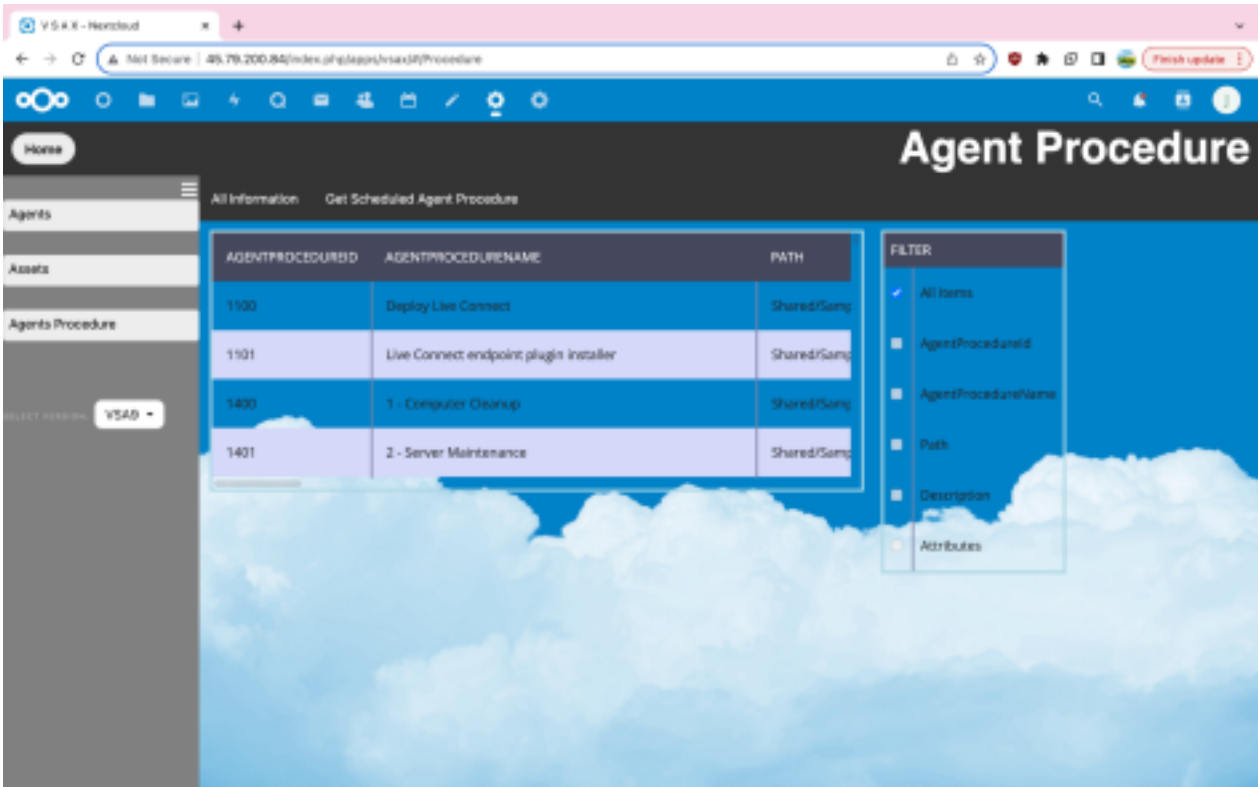
Appendix A - UML Diagrams

Final Deliverable Project Name





Appendix B - User Interface Design



VSAR - Herodotus

Not Secure | 45.79.200.84/index.php/apps/vsax09/VSAR_Assets

Assets

Home

Agents

Assets

Assets Procedure

SELECT VERSION VSAR

All Information Specific Agents

IDENTIFIER	NAME	GROUPNAME	DESCRIPTION
8D4c97c-88a0-4d20-80d5-4a7420878330	Nicholas's MacBook Pro	Default	MacOS 13.
b6a7dfbc-01f3-44c3-a01e-cab6d783e55c	Masoud's MacBook Pro	Default	MacOS 13.
c5b77979-0879-4bda-9c12-315709553b33	DESKTOP-QJ8QH06	Default	Windows 1
833ec15c-33a9-4ac8-9bd5-f77c7d5883c7	DESKTOP-TM6QQQE	Default	Windows 1

FILTER

- ☒ All Items
- ☐ Identifier
- ☐ Name
- ☐ GroupName
- ☐ Description
- ☐ Tags

VSAR - Herodotus

Not Secure | 45.79.200.84/index.php/apps/vsax09/Agents

Agents

Home

Agents

Assets

Agents Procedure

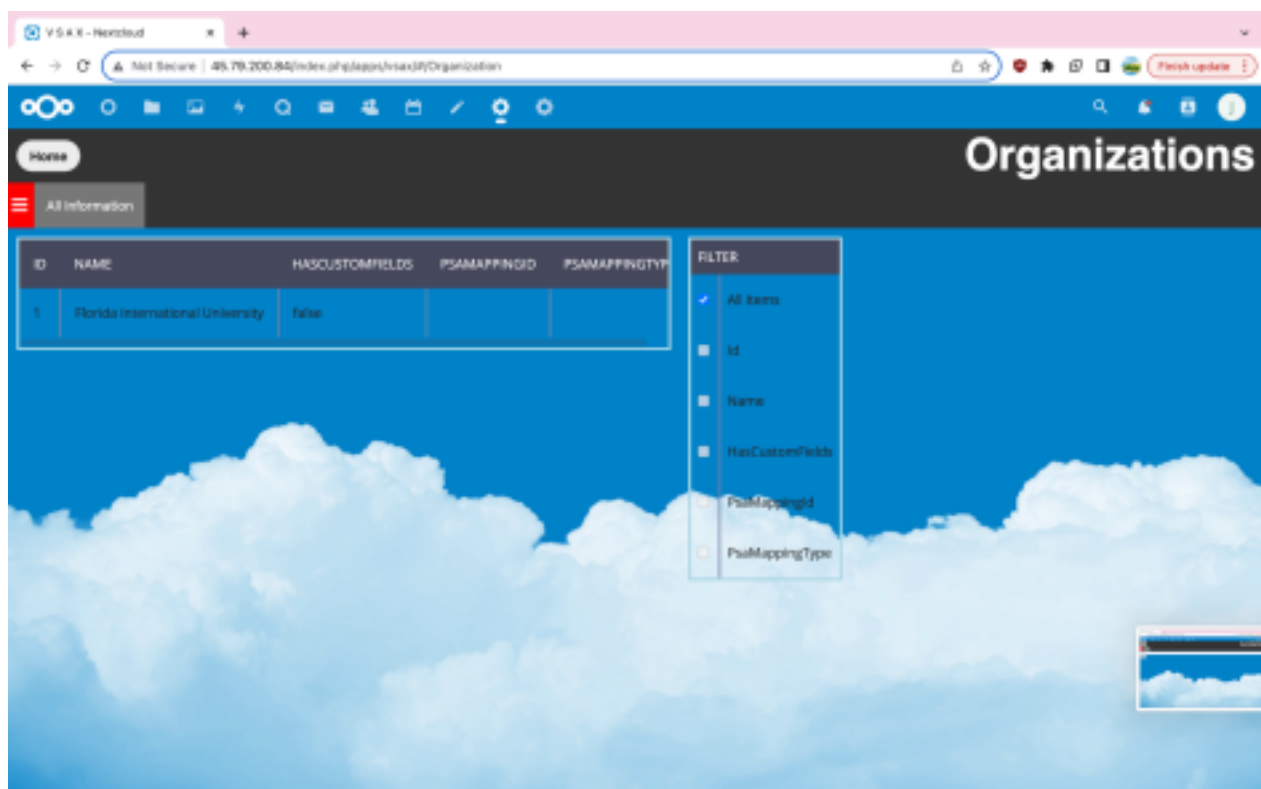
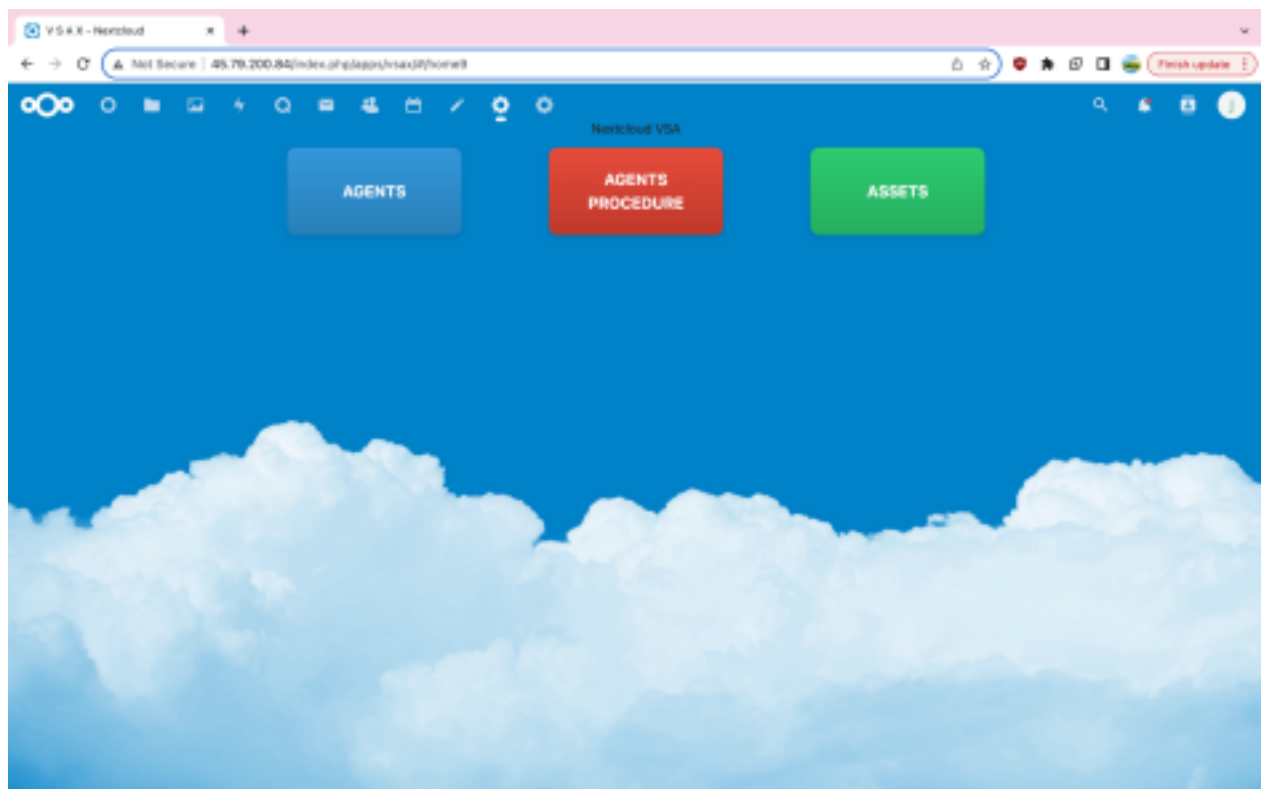
SELECT VERSION VSAR

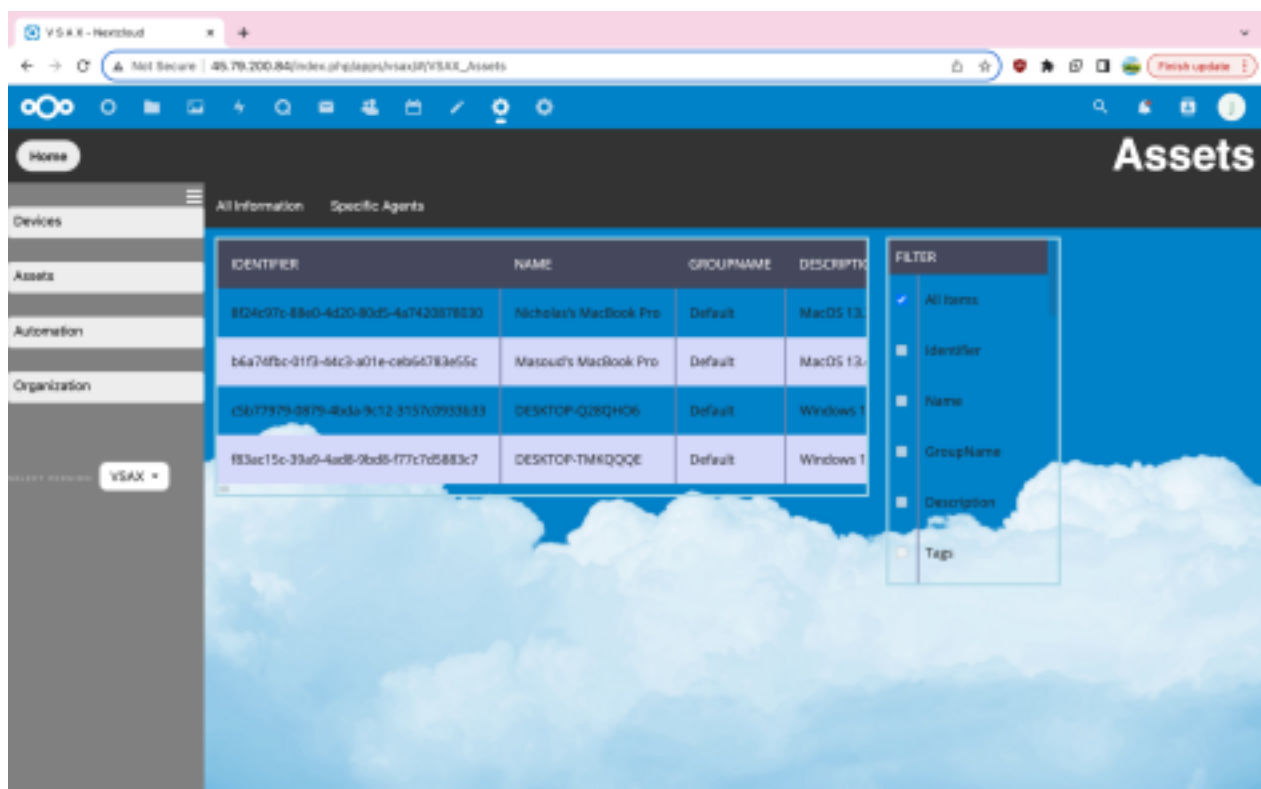
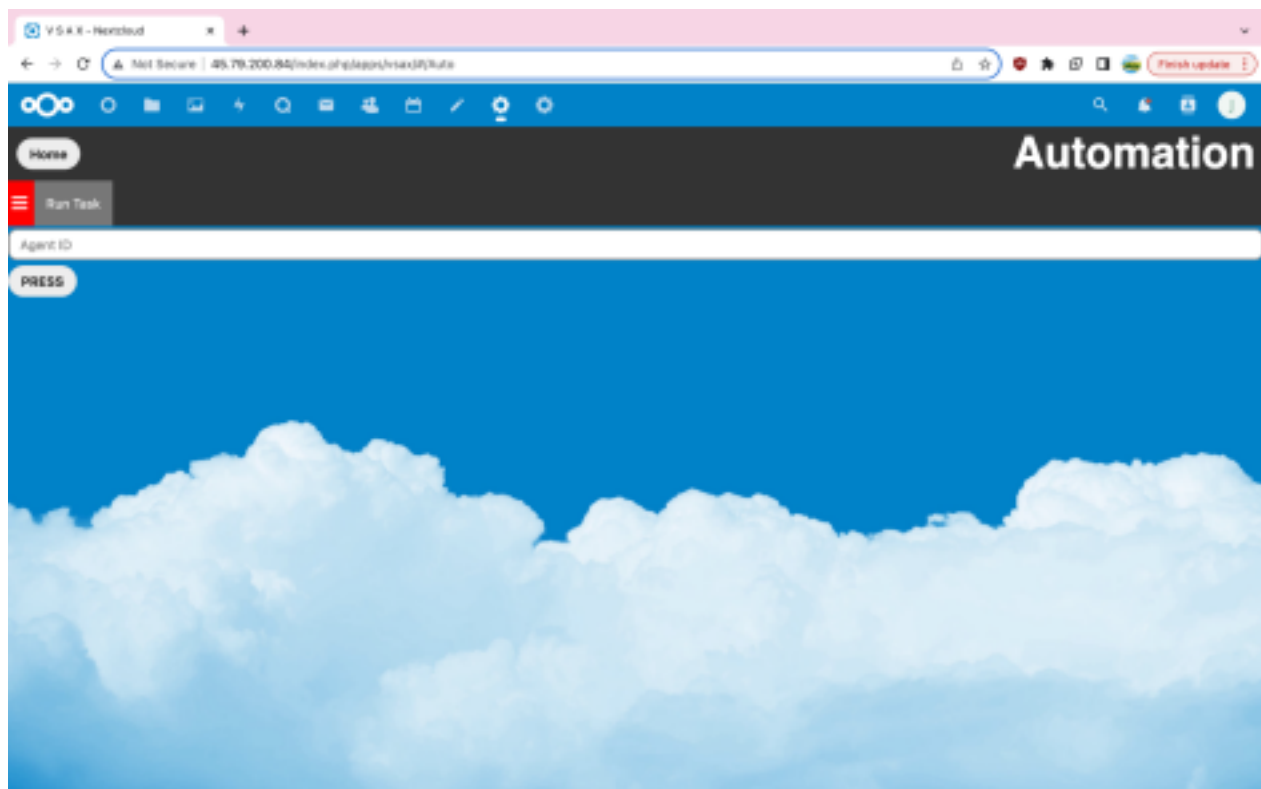
All Information Specific Agent Change Agent Name

AGENTID	AGENTNAME	ONLINE	OSTYPE	OSINFO
34770305804587	Alex.root.flu	0	10	Home x64 Edition Build 1
583163745410116	nicholas-mbp-3.root.flu	11	Mac OS X	Mac OS X 13.3.1 Build: 22
597673189481131	jules.root.flu	0	10	Professional x64 Edition 1

FILTER

- ☒ All Items
- ☐ AgentId
- ☐ AgentName
- ☐ Online
- ☐ OSType
- ☐ OSInfo





VSAX - Nextcloud

Not Secure | 45.79.200.84/index.php/apps/vsax/Devices

Devices

Home

Devices

All Information Specific Device

IDENTIFIER	NAME	GROUPNAME	GROUPID
8D4c97c-88e0-4d20-80d5-4a7423878230	Nicholas's MacBook Pro	Default	1
b6a7dfbc-01f3-44c3-a01e-cab6d783e55c	Masoud's MacBook Pro	Default	1
c5b77979-8d79-4bda-9c12-3157c9953b33	DESKTOP-QJ8QH06	Default	1
f83ec15c-33a9-4ac8-9bd5-f77c7d5883c7	DESKTOP-TM6QQQE	Default	1

Filter

- ☒ All Items
- ☐ Identifier
- ☐ Name
- ☐ GroupName
- ☐ GroupId
- ☐ IsAgentInstalled

VSAX - Nextcloud

Not Secure | 45.79.200.84/index.php/apps/vsax/home#

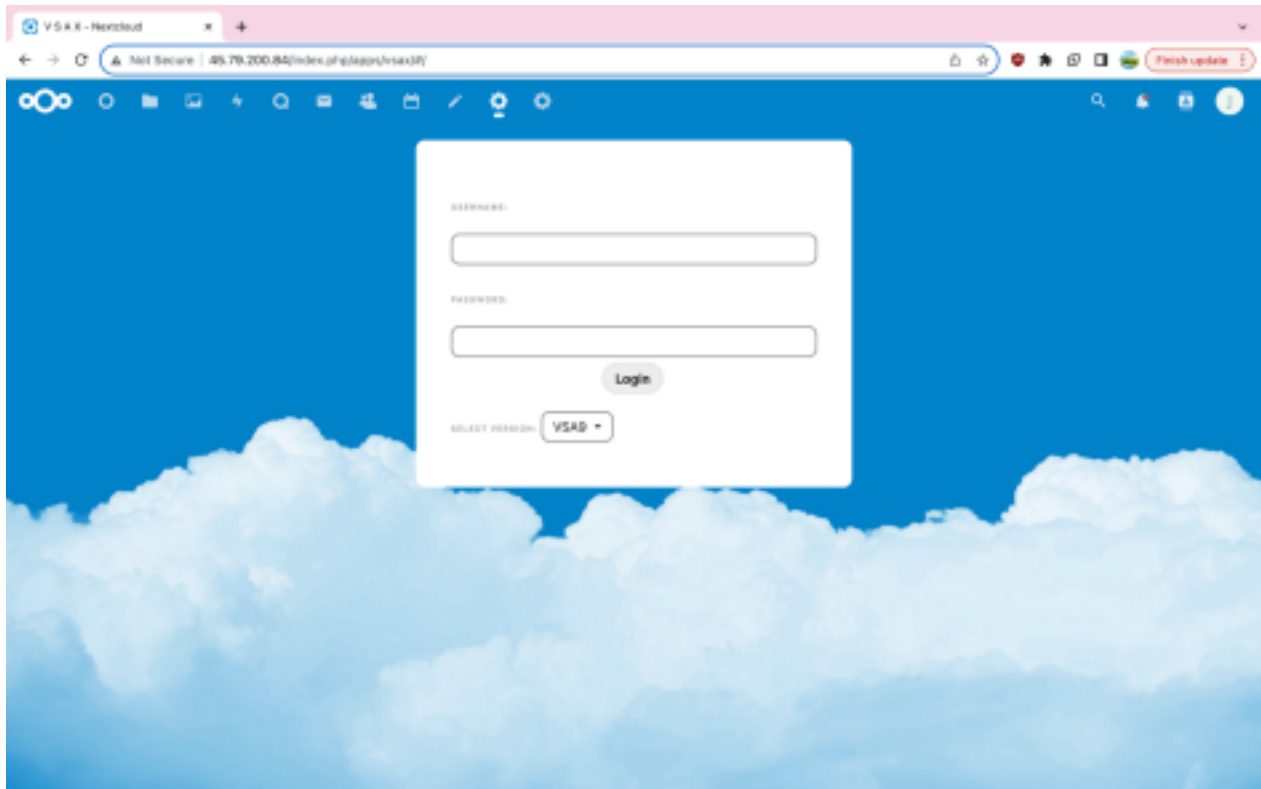
Nextcloud VSA

DEVICES

ASSETS

AUTOMATION

ORGANIZATION



Appendix C – Sprint Review Reports

Sprint Review 2 Meeting Minutes

Project: Nextcloud App: Exploring Integration with Kaseya VSA

Attendees: Julian Arias, Alexander Chirinos, Nicholas Samaroo

Start time: 6:30PM

End time: 7:00PM

After a show and tell presentation, the implementation of the following user stories was accepted by the product owners:
All.

- Deploy the agents to the Kaseya VSA

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

- What had to be moved back was the deployment of the agents

Sprint Review 3 Meeting Minutes

Project: Nextcloud App: Exploring Integration with Kaseya VSA

Attendees: Julian Arias, Alexander Chirinos, Nicholas Samaroo, Masoud Sadjadi

Start time: 12:00pm

End time: 12:30pm

After discussion, the velocity of the team was estimated to be <Enter the estimated team velocity>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- Figuring out the problem on deploying the Agents to Kaseya VSA
-

The team members indicated their willingness to work on the following user stories.

- Alexander is ready to start to deploy the agents to Kaseya VSA and implement Kaseya API into nextcloud
- Nicholas is ready to start to deploy the agents to Kaseya VSA and implement Kaseya API into nextcloud
- Julian is ready to start to deploy the agents to Kaseya VSA and implement Kaseya API into nextcloud

Sprint Review 4 Meeting Minutes

Project: Nextcloud App: Exploring Integration with Kaseya VSA

Attendees: Julian Arias, Alexander Chirinos, Nicholas Samaroo

Start time: 5:00 PM

End time: 5:30 PM

After discussion, the velocity of the team for the following spring was estimated to be <Enter the estimated team velocity>.

The product owner chose the following user stories to be done during the following sprint. They are ordered based on their priority.

- Continue to Study Kaseya VSA REST API
- Instead on focusing on the frontend we implement the REST API onto the nextcloud app

The team members indicated their willingness to work on the following user stories.

- Alexander will continue to practice Kaseya vsa REST API and implement the REST API onto the nextcloud app
- Nicholas will continue to practice Kaseya vsa REST API and implement the REST API onto the nextcloud app
- Julian will continue to practice Kaseya vsa REST API and implement the REST API onto the nextcloud app

Sprint Review 5 Meeting Minutes

Project: Nextcloud App: Exploring Integration with Kaseya VSA

Attendees: Julian Arias, Alexander Chirinos, Nicholas Samaroo, Masoud Sadjadi

Start time: 10:30

End time: 10:45

After discussion, the velocity of the team was estimated to be <Enter the estimated team velocity>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- To work on the frontend of the two apps and bring in more information like Agent Procedures for VSA 9

The team members indicated their willingness to work on the following user stories.

- Alexander is ready to work on the frontend
- Nicholas is ready to work on the frontend
- Julian is ready to work on the frontend

Sprint Review 6 Meeting Minutes

Project: Nextcloud App: Exploring Integration with Kaseya VSA

Attendees: Julian Arias, Alexander Chirinos, Nicholas Samaroo, Masoud Sadjadi

Start time: 10:00

End time: 10:30

After discussion, the velocity of the team was estimated to be <Enter the estimated team velocity>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- To fix a little on the frontend and fix any bugs to it

The team members indicated their willingness to work on the following user stories.

- Alexander is ready to work on the frontend
- Nicholas is ready to work on the frontend
- Julian is ready to work on the frontend

Sprint Review 7 Meeting Minutes

Project: Nextcloud App: Exploring Integration with Kaseya VSA

Attendees: Julian Arias, Alexander Chirinos, Nicholas Samaroo

Start time: 10:00

End time: 10:30

After discussion, the velocity of the team was estimated to be <Enter the estimated team velocity>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- Will start making the poster and other documentation

The team members indicated their willingness to work on the following user stories.

- Alexander is ready to work on the poster and other documentation
- Nicholas is ready to work on the poster and other documentation
- Julian is ready to work on the poster and other documentation

Appendix D - User Manuals, Installation/Maintenance Document,

Shortcomings/Wishlist Document and other documents

User Manual

Navigating through our project is a straightforward and user-friendly experience. Begin by logging in using either your VSA 9 or VSA 10 credentials on the login screen. The interface offers the convenience of switching between the two versions directly from this login screen, providing flexibility based on your specific needs.

Upon logging in with VSA 9 credentials, you'll encounter a set of distinct buttons: Agents, Agents Procedure, and Assets. These buttons serve as portals to dedicated pages where you can access and review relevant information. For example, clicking on the "Agents" button will seamlessly transition you to the Agents page, offering a comprehensive view of agent-related data. The same principle applies to the "Agents Procedure" and "Assets" buttons, each directing you to pages specifically tailored to present pertinent details in an organized manner.

To facilitate smooth navigation, a home button is strategically placed at the top left corner of the interface. This feature allows users to swiftly return to the main screen from any subsection, maintaining an intuitive browsing experience. Additionally, a navigation button is incorporated, offering a convenient overview of the available pages. Upon clicking, a menu displaying the two other pages will appear, enabling quick access to different sections of the application.

At the bottom of the interface, a user-friendly option is provided for seamless version switching between VSA 9 and VSA 10. This feature ensures that users can effortlessly explore and compare the functionalities of each version, tailoring their experience to specific requirements.

Whether you're focused on agent management, procedural details, or asset tracking, our project's intuitive design and navigation options aim to enhance user efficiency and satisfaction throughout the exploration process

Installation/Maintenance

Step 1:

- Install git on your computer <https://git-scm.com/downloads>
- Install Node.js (comes with npm) <https://nodejs.org/en/>

Step 2:

Download Visual Studio Code from your browser through:

<https://code.visualstudio.com/download>

Building the App

You must first have a nextcloud server that you have access to.

After you download all the files, and if you used the zip download, you unzip the folder and place it into your nextcloud apps directory. For example my nextcloud directory was `/var/www/nextcloud/apps/`

In visual studio code, open the terminal and cd into the folder of the project and type

npm install.

Changing some of the Code

cd into the backend folder inside the project.

In the code Replace the localhost:4000 with your own nextcloud server IP address.

In the src folder and inside the VSA9_views and VSAX_views folder change everything that says localhost:4000/ to the nextcloud server address

Running the App

Cd into the backend folder and type npm start, to run the server side.

The frontend should already be up when placing the project folder into the nextcloud apps directory.

