MICHIGAN STATE
UNIVERSITY

Project Plan
SmartSat™ Satellite App Store

The Capstone Experience

Team Lockheed Martin Space
Customer: Josh Davidson, Software Engineer

Brian Fuessel
Daniel Webb
Peng Sun
Sailesh Gundepudi
Tony Miller
Department of Computer Science and Engineering
Michigan State University

Spring 2020

From Students…
…to Professionals

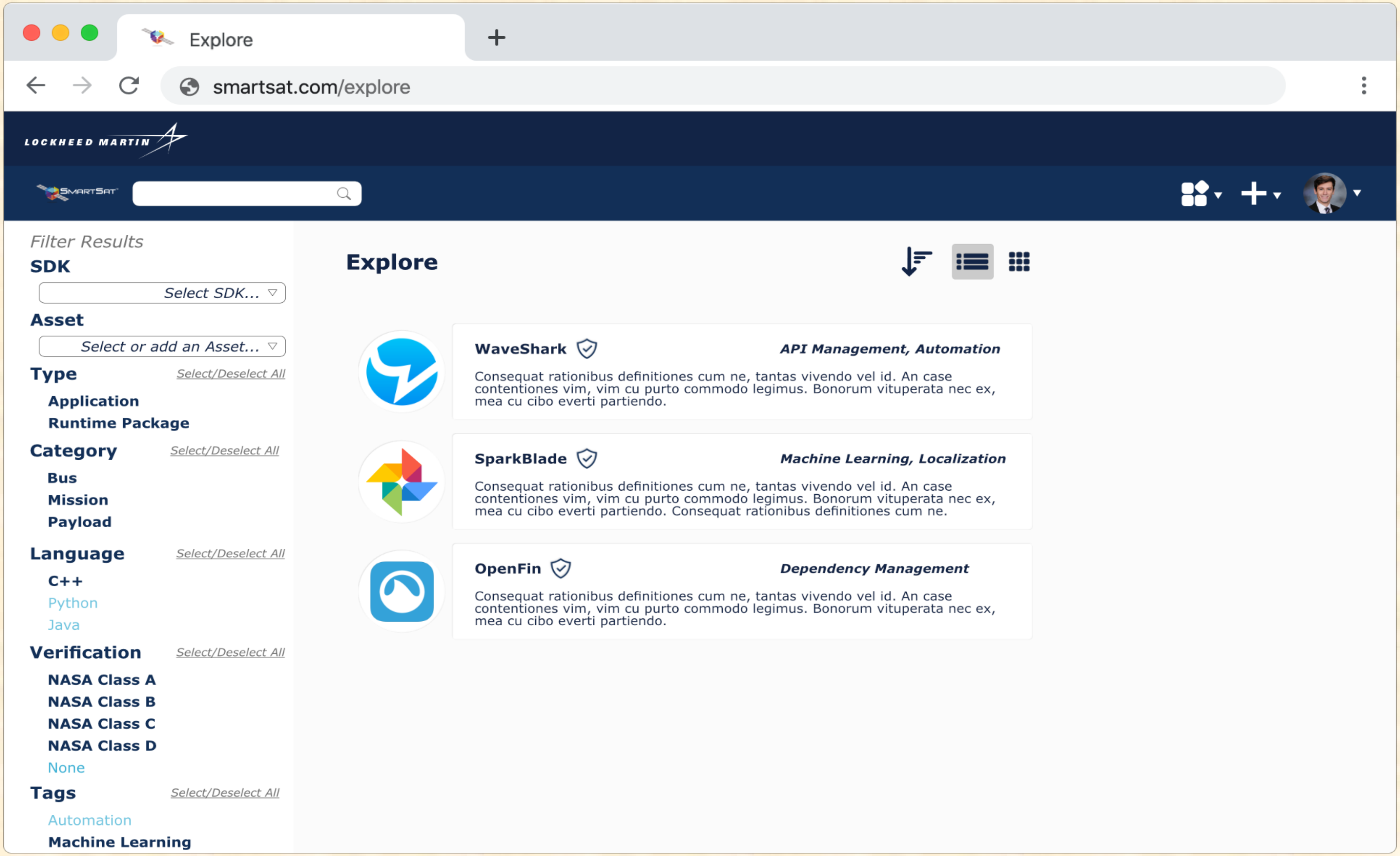
Functional Specifications

- Lockheed Martin SmartSat™ applications expand the capabilities of satellites
- Clients, developers require a way to manage and browse existing applications
- SmartSat™ App Store acts as the central access point for satellite applications
- Users can upload and download applications with ease

Design Specifications

- A marketplace for satellite applications
- Users can upload, download, update, and deploy these applications – both locally and to a live satellite
- When applications are uploaded, they are tested against an array of satellite configurations to check for compatibility
- Applications can be verified by specific users, to show that they meet certain standards

Screen Mockup: Explore



Filter Results

SDK

Select SDK... ▽

Asset

Select or add an Asset... ▽

Type *Select/Deselect All*

- **Application**
- **Runtime Package**

Category *Select/Deselect All*

- **Bus**
- **Mission**
- **Payload**

Language *Select/Deselect All*

- **C++**
- Python
- Java

Verification *Select/Deselect All*

- **NASA Class A**
- **NASA Class B**
- **NASA Class C**
- **NASA Class D**
- None

Tags *Select/Deselect All*

- Automation
- **Machine Learning**

Explore

WaveShark — ***API Management, Automation***

Consequat rationibus definitiones cum ne, tantas vivendo vel id. An case contentiones vim, vim cu purto commodo legimus. Bonorum vituperata nec ex, mea cu cibo everti partiendo.

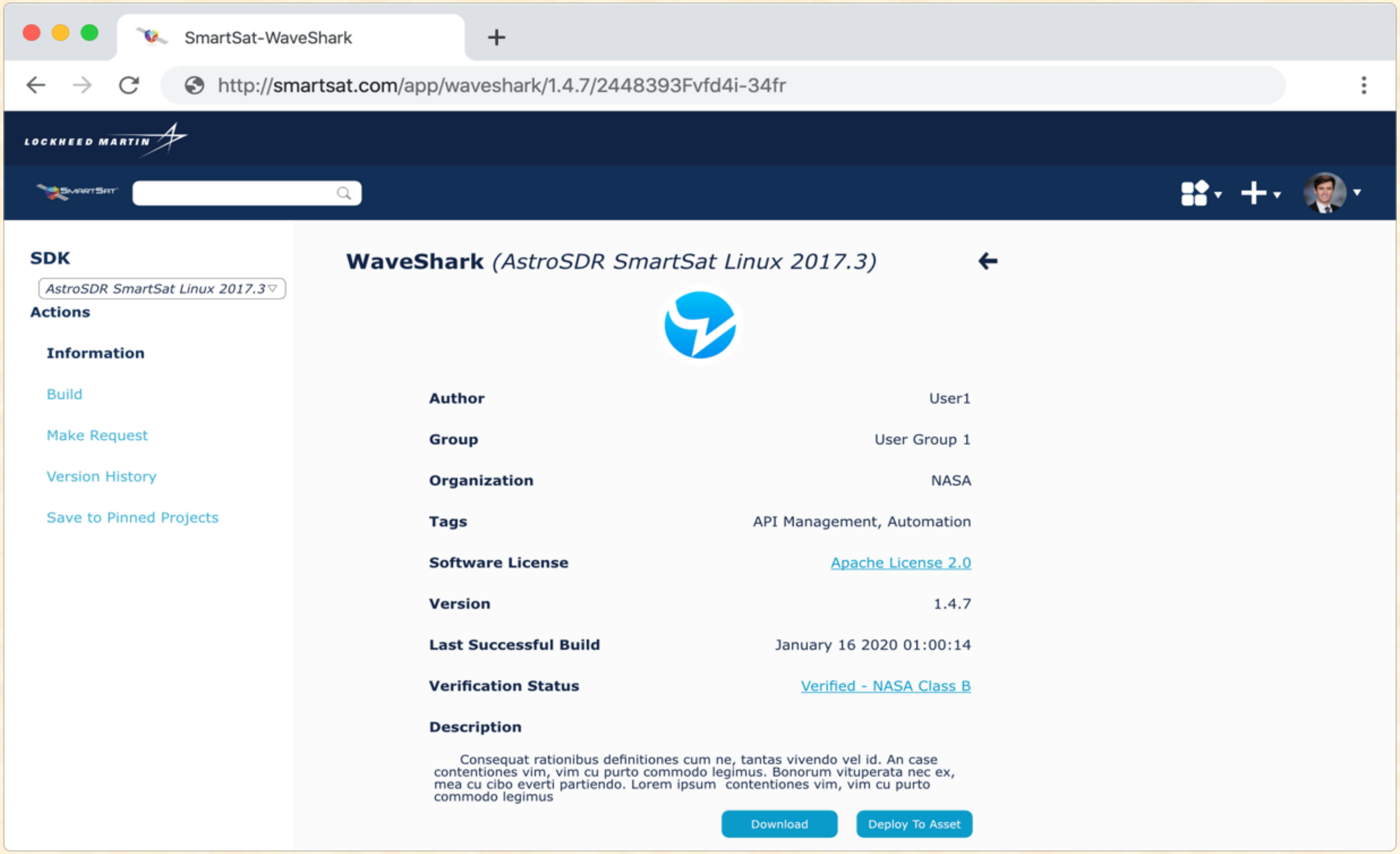
SparkBlade — ***Machine Learning, Localization***

Consequat rationibus definitiones cum ne, tantas vivendo vel id. An case contentiones vim, vim cu purto commodo legimus. Bonorum vituperata nec ex, mea cu cibo everti partiendo. Consequat rationibus definitiones cum ne.

OpenFin — ***Dependency Management***

Consequat rationibus definitiones cum ne, tantas vivendo vel id. An case contentiones vim, vim cu purto commodo legimus. Bonorum vituperata nec ex, mea cu cibo everti partiendo.

Screen Mockup: Application Page



SDK

AstroSDR SmartSat Linux 2017.3 ▽

Actions

Information

Build

Make Request

Version History

Save to Pinned Projects

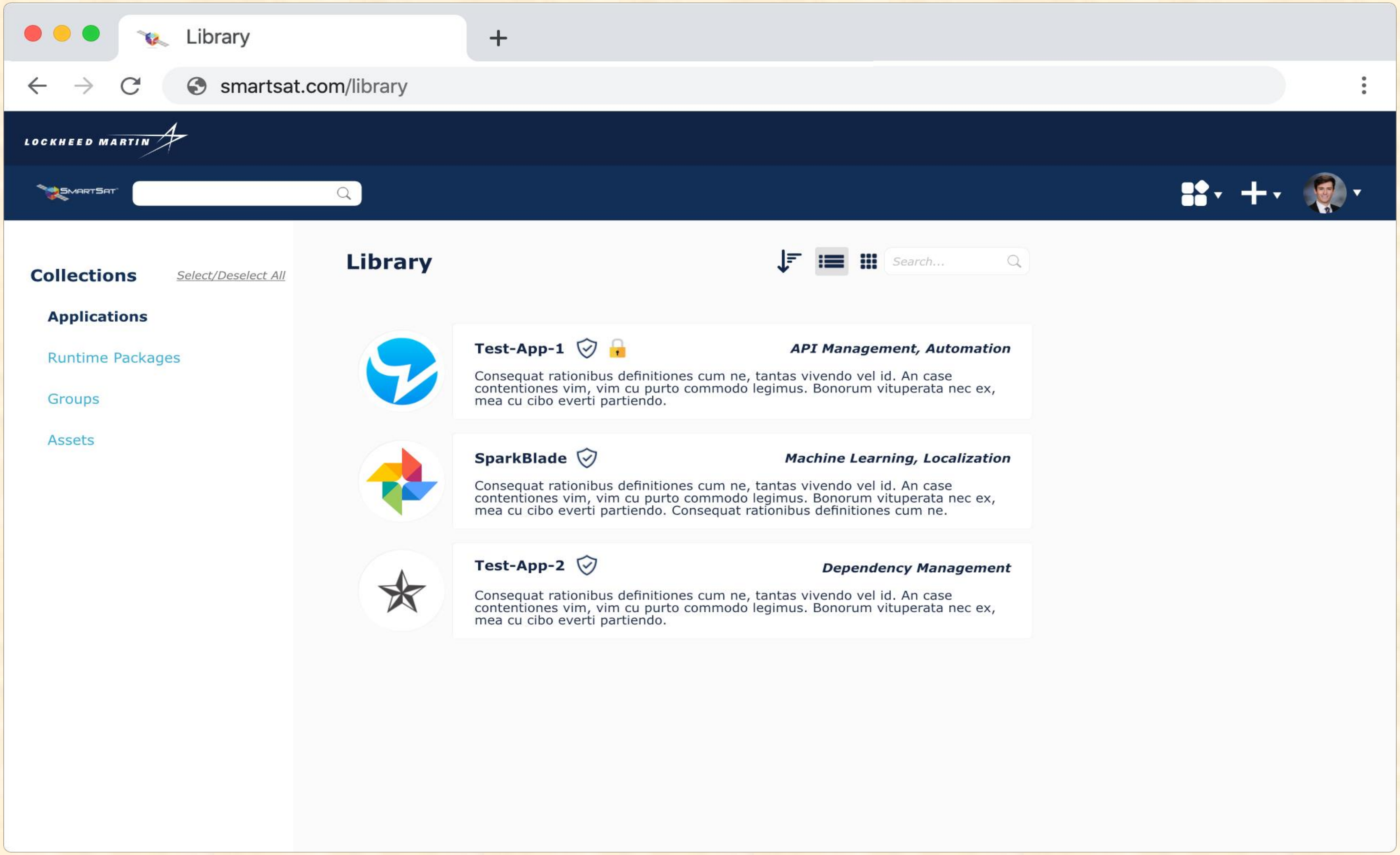
WaveShark *(AstroSDR SmartSat Linux 2017.3)*

Author	User1
Group	User Group 1
Organization	NASA
Tags	API Management, Automation
Software License	Apache License 2.0
Version	1.4.7
Last Successful Build	January 16 2020 01:00:14
Verification Status	Verified - NASA Class B

Description

Consequat rationibus definitiones cum ne, tantas vivendo vel id. An case contentiones vim, vim cu purto commodo legimus. Bonorum vituperata nec ex, mea cu cibo everti partiendo. Lorem ipsum contentiones vim, vim cu purto commodo legimus

Screen Mockup: User Library



Library

Collections *Select/Deselect All*

- **Applications**
- Runtime Packages
- Groups
- Assets

Test-App-1 — ***API Management, Automation***

Consequat rationibus definitiones cum ne, tantas vivendo vel id. An case contentiones vim, vim cu purto commodo legimus. Bonorum vituperata nec ex, mea cu cibo everti partiendo.

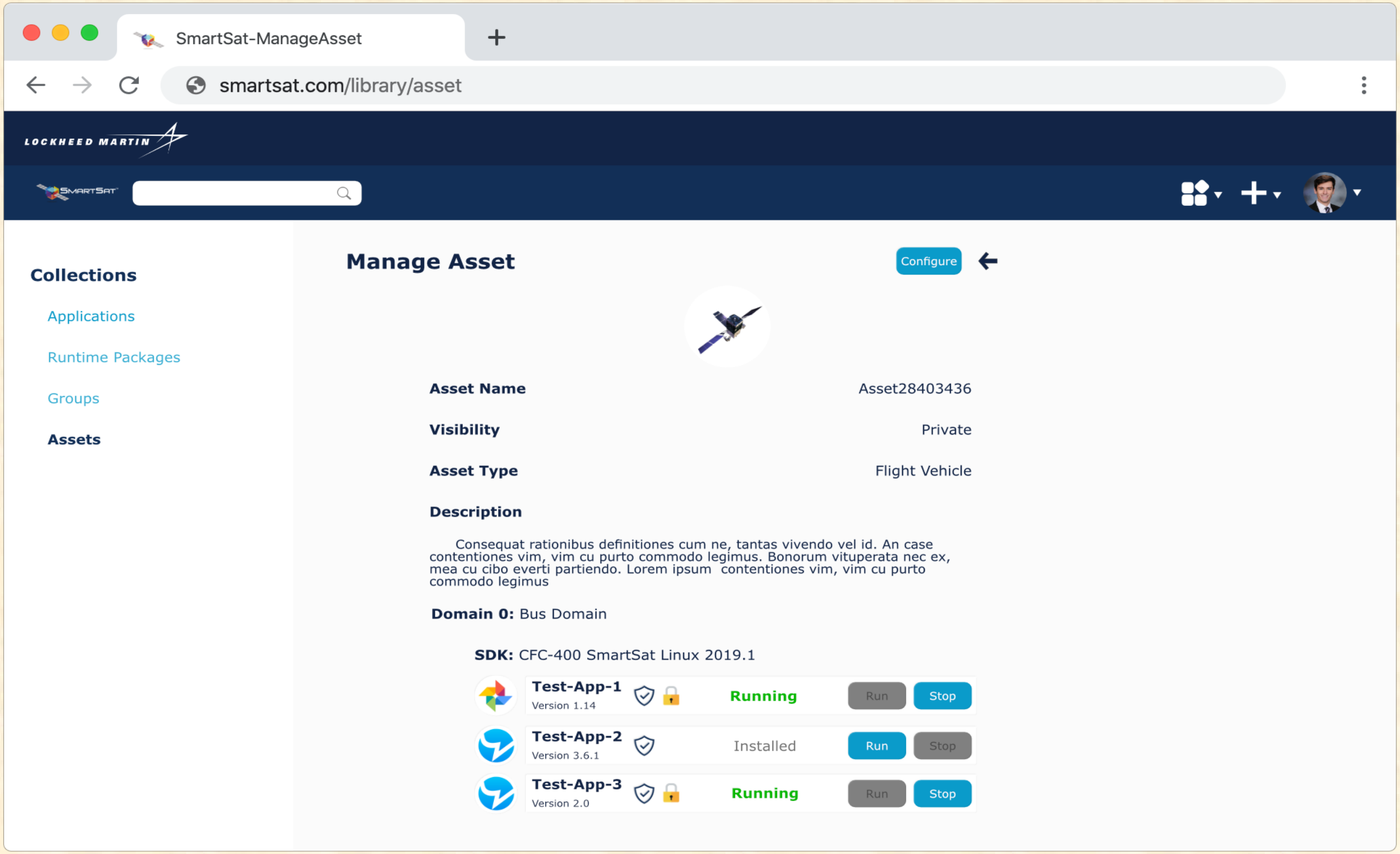
SparkBlade — ***Machine Learning, Localization***

Consequat rationibus definitiones cum ne, tantas vivendo vel id. An case contentiones vim, vim cu purto commodo legimus. Bonorum vituperata nec ex, mea cu cibo everti partiendo. Consequat rationibus definitiones cum ne.

Test-App-2 — ***Dependency Management***

Consequat rationibus definitiones cum ne, tantas vivendo vel id. An case contentiones vim, vim cu purto commodo legimus. Bonorum vituperata nec ex, mea cu cibo everti partiendo.

Screen Mockup: User Asset Page



Manage Asset

Asset Name	Asset28403436
Visibility	Private
Asset Type	Flight Vehicle

Description

Consequat rationibus definitiones cum ne, tantas vivendo vel id. An case contentiones vim, vim cu purto commodo legimus. Bonorum vituperata nec ex, mea cu cibo everti partiendo. Lorem ipsum contentiones vim, vim cu purto commodo legimus

Domain 0: Bus Domain

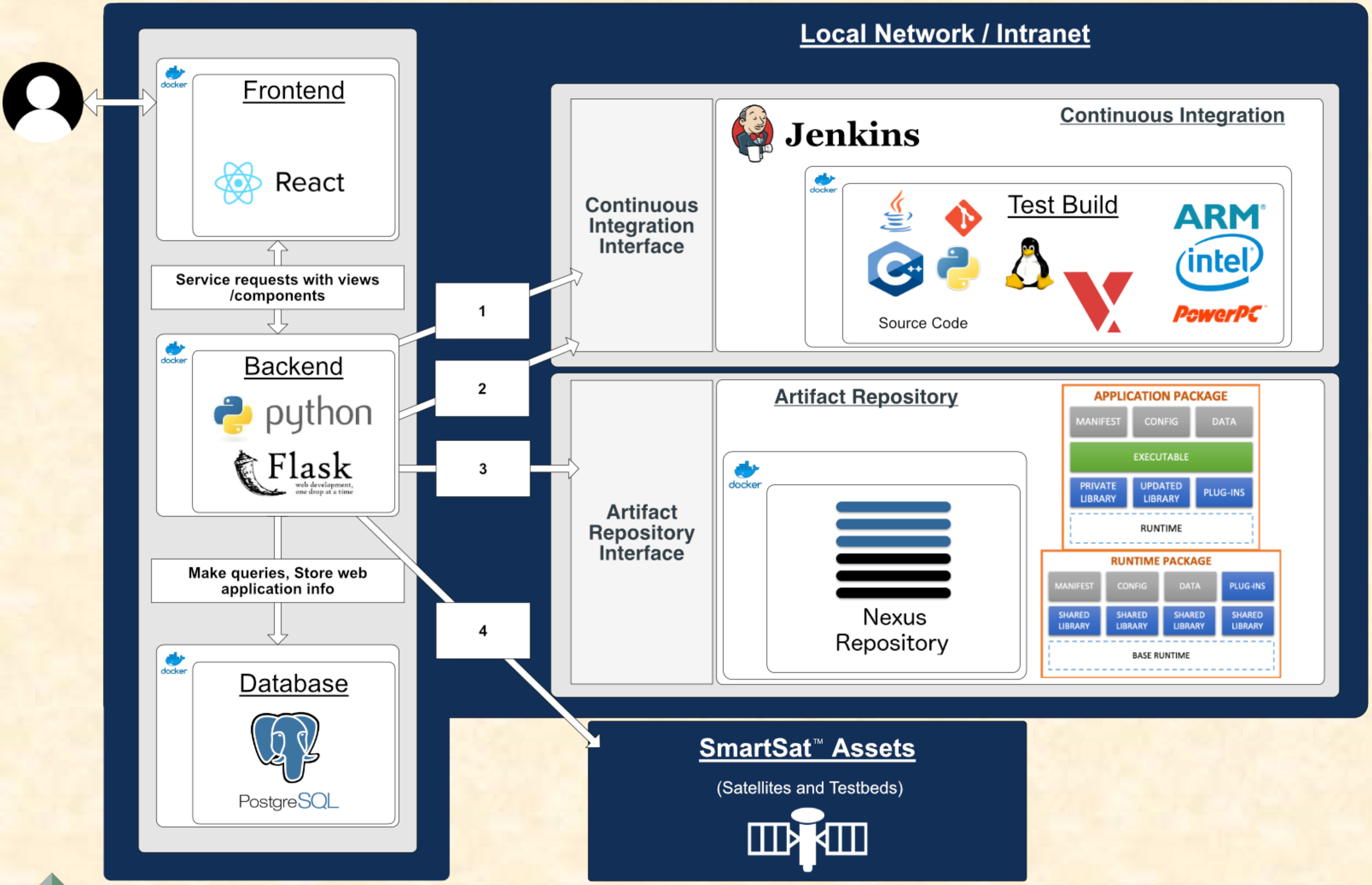
SDK: CFC-400 SmartSat Linux 2019.1

Test-App-1 Version 1.14	Running	Run	Stop
Test-App-2 Version 3.6.1	Installed	Run	Stop
Test-App-3 Version 2.0	Running	Run	Stop

Technical Specifications

- Continuous Integration Server rebuilds apps when new SDKs are uploaded
- Nexus repository manager stores project artifacts
- Users can deploy/manage applications on an array of Lockheed Martin assets
- Upload apps in source or binary form
- Download apps locally in chosen format (tar.gz, git link)

System Architecture



Local Network / Intranet

Frontend

React

Service requests with views /components

Backend

python

Flask

Make queries, Store web application info

Database

PostgreSQL

Continuous Integration Interface

Continuous Integration

Jenkins

Test Build

Source Code

Artifact Repository Interface

Artifact Repository

Nexus Repository

APPLICATION PACKAGE

MANIFEST, CONFIG, DATA, EXECUTABLE, PRIVATE LIBRARY, UPDATED LIBRARY, PLUG-INS, RUNTIME

RUNTIME PACKAGE

MANIFEST, CONFIG, DATA, PLUG-INS, SHARED LIBRARY, SHARED LIBRARY, SHARED LIBRARY, SHARED LIBRARY, BASE RUNTIME

SmartSat™ Assets

(Satellites and Testbeds)

Legend

1

Submit requests for new builds (uploading a new version)

2

Retrieve results for each build (Report failure info, update version, app info upon success)

3

Archive successful builds

4

Deploy applications to live vehicles

System Components

- Hardware Platforms
  - NVIDIA Jetson TX2
  - ZYNQ UltraScale+
  - Lockheed Martin Assets
- Software Platforms / Technologies
  - Front End: ReactJS
  - Back End: Flask (Python) with PostgreSQL database
  - Continuous Integration: Jenkins CI
  - Artifact Repository: Nexus
  - Containerization: Docker

Risks

- Risk 1
  - Proprietary App/Project Dependencies
  - Identify problems early as possible to give us enough time to report to the client, and work through/adjust accordingly
- Risk 2
  - Deploying Foreign Software to Unfamiliar Hardware
  - Begin testing and deploying rudimentary builds this week to highlight potential roadblocks & understand what has already been implemented
- Risk 3
  - Continuous Integration Server Implementation
  - Researched common CI solutions (Jenkins, etc.) last week, working with Jenkins more in-depth this week
- Risk 4
  - Security of Confidential Project Data
  - Secure data transmission channels have been set up (last week)

Questions?

? ? ? ?

? ?

? ? ?