# Use cases

## System administrator installs FOSSology-Ninka

1. **Title:** System administrator installs FOSSology-Ninka
2. **Primary Actor:** System administrator on behalf of users who generate SPDX documents
3. **Goal in Context:** To speed the determination of SPDX licensing data by comparing the output of multiple automatic software license scanners
4. **Stakeholders and Interests:**
   1. **System administrators:**
      1. To provide users with the tools they need for their tasks
      2. To ensure the availability and responsiveness of user applications
   2. **Scanning users:** To speed the determination of SPDX data by comparing the output of multiple automatic software license scanners
5. **Preconditions:**
   1. Installation of FOSSology and Ninka with appropriate system resources to run them serially
   2. Installation of FOSSology-Ninka infrastructure components (e.g., Python interpreter)
6. **Main Success Scenario:**
   1. FOSSology-Ninka can successfully locate and execute FOSSology and Ninka
   2. FOSSology-Ninka is executable by authorized users
7. **Failed End Condition:**
   1. System administrator cannot determine how to configure FOSSology-Ninka to access its dependent systems (FOSSology, Ninka, Python, etc.)
   2. FOSSology-Ninka is not executable by authorized users
8. **Trigger:** User request for installation
9. **Notes:** This use case assumes that system administrators possess the requisite knowledge to install and configure FOSSology, Ninka, and Python for standalone operation (i.e., to function as expected in use cases other than FOSSology-Ninka).

## User scans package with FOSSology-Ninka through command line with output to the local file system

1. **Title:** User scans package with FOSSology-Ninka through command line with output to the local file system
2. **Primary Actor:** Users who generate SPDX documents
3. **Goal in Context:** To speed the determination of SPDX licensing data by comparing the output of multiple automatic software license scanners
4. **Stakeholders and Interests:**
   1. **Scanning users:**
      1. To communicate the licensing information for their copyrightable artifacts, including all copyrightable artifacts they contain
      2. To speed the determination of SPDX data by comparing the output of multiple automatic software license scanners
   2. **Consumers of files and packages:**
      1. To receive accurate and clear information of licensing of artifacts
      2. To be able to comply easily with licenses for artifacts
      3. To be able to trust that the package SPDX data is in alignment with license assertions
5. **Preconditions:**
   1. Artifacts contain some licensing information (e.g., in comments)
   2. FOSSology and/or Ninka license determination algorithms are accurate
   3. The user is able to manually resolve conflicting license determinations
6. **Main Success Scenario:** Output allows the user to accurately and completely determine SPDX licensing data
   1. SPDX fields on which FOSSology and Ninka agree are filled out
   2. SPDX fields on which FOSSology and Ninka conflict are marked NO ASSERTION
   3. A data structure within the SPDX Comments field identifies conflict fields and lists the conflicting assertions
7. **Failed End Condition:**
   1. Failure to run FOSSology or Ninka, or
   2. Failure to compare FOSSology and Ninka output, or
   3. Failure to create an intermediate SPDX document on the local file system
8. **Triggers:**
   1. User must determine licensing information for a file or package belonging to the user’s organization
   2. User must determine licensing information for a file or package that the user’s organization would like to use
9. **Notes:** FOSSology-Ninka output alone does not guarantee accurate and complete determination of SPDX licensing data. Artifact licensing information may be incomplete; FOSSology or Ninka may be inaccurate; the user may be unable to manually resolve conflicting assertions. FOSSology-Ninka should still speed the determination process over purely manual review.
10. **Example:** A user is tasked with determining the licenses used in a third-party software package. The user inputs the package to FOSSology-Ninka, which outputs a provisional SPDX document. The document asserts that the license is GPLv3 and does not contain conflict information, indicating that FOSSology and Ninka agree.

## User scans package with FOSSology-Ninka through command line with output to an instance of the SPDX Dashboard web application

1. **Title:** User scans package with FOSSology-Ninka through command line with output to an instance of the SPDX Dashboard web application
2. **Primary Actor:** Users who generate SPDX documents
3. **Goal in Context:** To speed the determination of SPDX licensing data by comparing the output of multiple automatic software license scanners
4. **Stakeholders and Interests:**
   1. **Scanning users:**
      1. To communicate the licensing information for their copyrightable artifacts, including all copyrightable artifacts they contain
      2. To speed the determination of SPDX data by comparing the output of multiple automatic software license scanners
   2. **Dashboard users:**
      1. **TODO**
   3. **Consumers of files and packages:**
      1. To receive accurate and clear information of licensing of artifacts
      2. To be able to comply easily with licenses for artifacts
      3. To be able to trust that the package SPDX data is in alignment with license assertions
5. **Precondition:** Artifacts contain some licensing information (e.g., in comments)
6. **Main Success Scenario:** Output allows the user to accurately and completely determine SPDX licensing data
   1. SPDX fields on which FOSSology and Ninka agree are filled out
   2. SPDX fields on which FOSSology and Ninka conflict are marked NO ASSERTION
   3. A data structure within the SPDX Comments field identifies conflict fields and lists the conflicting assertions **TODO**
7. **Failed End Condition:**
   1. Failure to run FOSSology or Ninka, or
   2. Failure to compare FOSSology and Ninka output, or
   3. Failure to create an intermediate SPDX document and upload it to the Dashboard
8. **Triggers:**
   1. User must determine licensing information for a file or package belonging to the user’s organization
   2. User must determine licensing information for a file or package that the user’s organization would like to use
9. **Notes:** FOSSology-Ninka output alone does not guarantee accurate and complete determination of SPDX licensing data. Artifact licensing information may be incomplete; FOSSology or Ninka may be inaccurate; the user may be unable to manually resolve conflicting assertions. FOSSology-Ninka should still speed the determination process over purely manual review. **TODO**
10. **Example:** A user is tasked with determining the licenses used in a third-party software package. The user inputs the package to FOSSology-Ninka, which outputs a provisional SPDX document. The document asserts that the license is GPLv3 and does not contain conflict information, indicating that FOSSology and Ninka agree. **TODO**

# Changelog

**TODO**