

Capstone Weekly Status Report 9

Date and Time: November 2, 2025

Student Name: Satwik Alla

Capstone Project: SBOM Manager - Software Bill of Materials Management System

Capstone Sponsor: Software Engineering Department **Ethical Software Lab** – In-house project

Capstone Advisor: Prof. Mohammad Samarah & Prof. Melissa Sahl

Status Details:

Work accomplished this Week

File Upload & SBOM Visualization Implementation:

File Upload Interface: Developed drag-and-drop file upload component with real-time progress tracking, file type validation (ZIP, APK, IPA, EXE, DEB, RPM), 50MB size limit enforcement, and status indicators (uploading/processing/completed/failed). Integrated FormData API for multipart uploads.

- **Application List View:** Built paginated application table with sorting (date, name, platform/OS filtering, search by application name, and responsive card/table layouts. Implemented infinite scroll for mobile view with lazy loading.
- **SBOM Detail Viewer:** Created comprehensive SBOM detail page displaying metadata (platform, version, file hash, upload timestamp), component hierarchical tree with expand/collapse functionality, license distribution pie charts using Recharts, and export functionality for CycloneDX JSON, SPDX JSON, and CSV formats.
- **Component Table:** Implemented sortable component table with columns (name, version, license, category, vendor), expandable rows showing dependency chains, search/filter within 500+ components, and pagination (25/50/100 rows per page).
- **Real-time Processing Updates:** Integrated polling mechanism (every 2 seconds) to track SBOM generation status, toast notifications on completion/failure, optimistic UI updates, and WebSocket connection for instant updates (implemented as stretch goal).

- **Backend Integration:** Connected all frontend components to FastAPI endpoints, implemented error handling with retry logic for failed uploads, added request cancellation for unmounted components, and optimized API calls using TanStack Query's caching.

Work Planned for Next Week

Search & Comparison Features:

Global Search: Build comprehensive search across applications, components, licenses, and versions with fuzzy matching, filters (date range, platform, license type), search history, and saved searches.

- **SBOM Comparison:** Implement side-by-side comparison view for 2+ applications, highlight added/ removed/modified components, license change detection, and visual diff with color coding.
- **Statistics Dashboard:** Create analytics page with charts (total applications, components by category, license breakdown, platform distribution), trend analysis over time, and export statistics as PDF reports.
- **Performance Optimization:** Add infinite scroll pagination for large component lists, implement virtual scrolling for 1000+ row tables, and optimize search queries with debouncing.
- **Advanced Filtering:** Build complex filter builder (AND/OR logic), filter by component vendor, filter by license risk level, and save custom filter presets.

Problems, obstacles, needs, or questions that I need help with from client, advisor, or instructor

- **WebSocket Reliability:** WebSocket connection drops on Render.com free tier after 5 minutes of inactivity. Should we fall back to polling for Alpha or upgrade hosting?
- **Export Performance:** Exporting SBOMs with 500+ components to CSV takes 3-4 seconds. Should we implement server-side export generation or optimize client-side processing?
- **Search Performance:** Fuzzy search across 10,000+ components cause UI lag. Should we implement backend search endpoint or use Web Workers for client-side search?
- **Comparison UX:** Comparing more than 2 applications becomes cluttered. Should we limit to 2 for Alpha or implement carousel/tabs for multiple comparisons?
- **Data Retention:** No policy defined for storing uploaded files. Should we auto-delete applications after 30 days or keep indefinitely for Alpha?

Overall Status

- Project Status Color: Green
- Project Status Color from Previous Week: Green