

EMSE 6801

Homework #3

Topic: Systems Engineering Modeling. With this assignment, you will experience a systems engineering design tool and see what various SE concepts look like in the form of conceptual models.

1. Install the Academic version of Vitech's CORE software (20 mins)
2. Locate the CORE Guided Tour and with your version of the software follow the guided tour to its completion in Chapter 7 (about 3 hours total). This will build a number of model elements describing a geospatial product library system.
3. Oops, we forgot cyber. We are going to have to add some representative Requirements, Functions, Items, Components, and Links to fix this. Feel free to add your own ideas too.
4. Add these new requirements to the requirements hierarchy setting appropriate attributes (e.g., name, number, type, description), some may require breaking down into smaller requirements, some may require corrections:
 - 3.1.8 Security Auditing – The system shall provide the ability to record and report on system and user events relative to system security. Rationale: customer security accreditation requires a security event auditing capability to meet local cyber security guidelines. Child requirements:
 - The system shall record all failed login attempts.
 - The system really should timestamp all auditable events and if an operator was involved, the operator's ID.
 - The system shall record all security violations attempted by the user.
 - The system shall maintain a time-indexed audit log.
 - The system shall generate a time-boxed audit report of selected system event types.
 - The system shall provide the means to allow a privileged operator to select which archive events are considered audit significant.
 - The system shall prevent all users from altering recorded events.
 - The system shall periodically back up and archive the active audit log and begin a fresh audit log.
 - The system shall complete an audit dump within TBD seconds of initiation.
5. Add a new (physical) component under the Geospatial Library component called the **Security Audit System**. It would be parallel to the Workstation and Command Center components. This should automatically add a **Perform Security Audit System Functions** function to the behavioral model. If it doesn't, create such a function and allocate it to the Security Audit System.

6. Place the **Perform Security Audit System Functions** function into the existing behavioral model – perhaps as a sub-function of Perform Geospatial Library Functions. Remember, it is always auditing so it needs to run in parallel with other Geospatial Library Functions.
7. Decompose the Perform Security Audit System Functions function with 5 – 10 sub-functions (or sub-sub-functions) to characterize its behavior. Things like these are appropriate - be creative:
 - Open Audit Log
 - Generate timestamp
 - Record login failures
 - Record security violations
 - Record user actions
 - Record key archive events
 - Dump Audit Log
 - Close Audit Log
8. Connect these functions with typical Item flows and Triggers to other functions
9. Generate any new requirements these new functions add using <based on> or trace them to existing requirements.
10. Establish a Link between the Security Audit System and the Geospatial Library transferring user and system activity information to the Security Audit System and audit reports back from the Security Audit System to the Geospatial Library.
11. Adjust any parts of the existing models to fit the Requirements, Functions, Components, and Links you added.
12. Make sure to continually save your work as you go.
13. When done, email me your .a90 file (CORE project archive)

Assessment: Completeness of the definition of new elements. Correctness in connecting the new elements to the existing models. Correctness of the new functional and physical flows.

Extra point getters:

- The addition of other functions, requirements, components, or links
- Experimentation with other elements (e.g., defined terms, risks, issues)
- Adding new use cases
- Experimentation with tool settings (e.g., colors, shapes, graphics)