OOSEM Report for System: Electron Microscope System Model

Phase 1: Setup Model

The first stage of the OOSEM method involves organizing the base model structure. For the Electron Microscope System, the model hierarchy is structured using SysML packages to cleanly separate design concerns.

- **Top-Level Model:** `Electron Microscope System Model`.
- **Package Structure:**
  - `Requirements`.
  - `DomainModel`.
  - `UseCases`.
  - `Subsystems`.

Phase 2: Analyze Stakeholder Needs

- **Project Stakeholders:**
  - **Acquisition/Funding (Who pays):** University Research Grants, Private Biotech Firms.
  - **Users/Beneficiaries (Who benefits):** Lab Technicians, Principal Investigators.
  - **Regulators/Constraints (Who constrains):** NIST (Standard Reference Materials).
- **Stakeholder Concerns:**
  - **Resolution:** The ability to visualize samples at sub-nanometer scales.
  - **Efficiency:** Rapid vacuum "pump down" times to increase sample throughput and prevent lab techs from losing time.
  - **Safety:** Automatic shutdown during a vacuum failure to protect the expensive electron source.
- **Enterprise Use Cases:** Derived directly from the stakeholder goals as active verb phrases.
  - `UC_001_Acquire_High_Resolution_Image` (Primary goal).
  - `UC_002_Calibrate_Electron_Beam` (Secondary goal).
  - `UC_003_Prepare_Specimen_Environment`.
  - `UC_004_Handle_Vacuum_Leak` (Exception/Rainy-Day path).
- **To-Be Domain BDD (Vocabulary Alignment):**
  - **System Domain Block:** `EMSystemDomain`.
  - **System of Interest:** `ElectronMicroscope`.

- **Actors in Domain:** [PLACEHOLDER] - *Specific SysML actor elements associated with the domain block on the To-Be BDD are undefined in the script.*
- **Domain Parts (Noun Catalog):** `ElectronSource`, `HighVoltagePowerSupply`, `VacuumChamber`, `VacuumPump`, `SpecimenStage`, `ElectromagneticLens`, `Detector`, `ImagingSoftware`, and `UserConsole`.

Phase 3: Analyze System Requirements

- **Mission Scenarios (Use Case Narratives):**
  - **Acquire High-Resolution Image (Main Success Scenario):** The technician loads the sample, the system evacuates the air, the beam is activated, and the image is captured.
  - **Handle Vacuum Leak (Exception Path):** If the vacuum pressure rises, the system immediately disables the high voltage and alerts the operator.
  - **Activity Diagram Actions & Object Flows:** [PLACEHOLDER] - *The specific SysML activity diagrams, control flows, and object tokens for these narratives are undefined.*
- **Capture Critical System Properties and Constraints:**
  - **Vacuum Threshold:** 10^{-5} Torr.
  - **Spatial Resolution:** $\le$ 50 picometers.
  - **Data Rate:** Minimum of 1 Gbps.
  - **Parametric Models:** [PLACEHOLDER] - *The specific Constraint Blocks, Constraint Properties, and Parametric Diagram binding connectors to enforce these mathematical constraints are undefined.*
- **Define System Context (IBD):**
  - [PLACEHOLDER] - *The System Context Internal Block Diagram (IBD), detailing the explicit flow properties, connectors, and full/proxy ports between the EM system and its external actors, is undefined.*
- **Specify Black-Box System Requirements (EARS Syntax):**
  - **Functional (REQ_001 - Vacuum Safety):** "If the vacuum pressure exceeds 10^{-5} Torr, then the system shall disable the Electron Source.".
  - **Performance (REQ_002 - Spatial Resolution):** "The system shall provide a spatial resolution of $\le$ 50 picometers.".
  - **Interface (REQ_003 - Detector Interface):** "The Detector shall transmit raw image data to the Controller at a minimum rate of 1 Gbps.".
- **Define System State Machine (Dynamic Behavior):**
  - **Target Subsystem:** `VacuumUnit` (`VacuumSM`).
  - **States:** `AtAtmosphere` (initial), `PumpingDown` (with a "Do" behavior: `RunVacuumPump`), and `OperationalVacuum`.

    - **Transitions:** The system accepts `startPumping` to transition from `AtAtmosphere` to `PumpingDown`, and utilizes a `PressureAchieved` signal to trigger the transition from `PumpingDown` to `OperationalVacuum`.
    - **Fault Resilience:** If power is lost, the system enters a `Safe Mode` utilizing a **History Pseudostate** to remember its status and resume venting safely.

Phase 4: Define Logical Architecture

This stage crosses the "Acquisition-Development Line" by grouping responsibilities into object-oriented logical subsystems rather than a rigid functional tree.

- **Logical Decomposition (Subsystem Blocks):**
    - **`IlluminationSubsystem`:** Composed of the `ElectronSource` and `HighVoltagePowerSupply`.
    - **`VacuumUnit` (Physical Environment):** Composed of the `VacuumChamber`, `VacuumPump`, and `SpecimenStage`.
    - **`ElectronColumn`:** Composed of the `ElectromagneticLens` (lenses) and `Detector`.
    - **`ControllerSubsystem`:** Composed of the `UserConsole` and `ImagingSoftware`.
- **Define Interaction Between Logical Components (Behavioral Allocation):**
    - **Illumination Action:** `GenerateBeam`.
    - **Vacuum Action:** `RunVacuumPump` (assigned to evacuate the chamber).
    - **Column Action:** `CaptureInteractions`.
    - **Controller Action:** `ProcessImageData`.
- **System Logical Architecture IBD:**
    - [PLACEHOLDER] - *The white-box Internal Block Diagram showing the internal reference associations, required/provided interfaces, and item flows (such as the 1 Gbps data link between the Detector and the ControllerSubsystem) is undefined.*

Phase 5-7: Physical Architecture, Traceability, & Optimization

- **Synthesize Candidate Physical Architectures:** [PLACEHOLDER] - *Specific physical hardware selection (e.g., vendor part numbers) was deliberately avoided to prevent over-modeling and ensure logical resilience.*
- **Manage Traceability:** Every element in the `SysML v2` model—from the 50 picometer requirement to the Vacuum State Machine—traces directly back to the core stakeholder concerns of image quality, lab efficiency, and safety. Detailed SysML `«trace»` or `«satisfy»` matrix implementations are [PLACEHOLDER].

- **Optimize and Evaluate Alternatives:** [PLACEHOLDER] - *Trade studies and alternative evaluations are undefined in this initial conceptual script.*

Here are the next chunks of the OOSEM conceptual design report, focusing on the formalization of the architecture into an executable system model and the definition of the resulting structural and behavioral views.

Phase 8: Model Formalization (SysML v2)

To bridge the gap between the conceptual design and the formal system architecture, the OOSEM outputs are codified into textual SysML v2. This provides an implementation-independent, executable specification of the system.

None

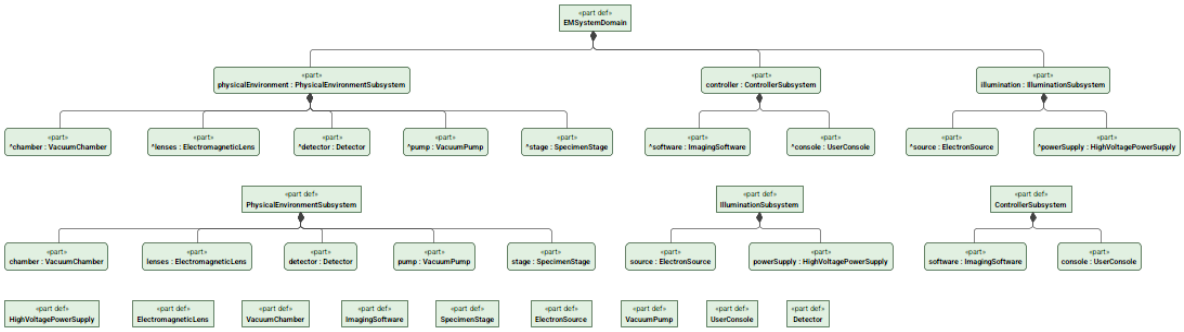
```
package 'Electron Microscope System Model' {

    //
-------------------------------------------------------------------
-------
    // 1) Requirements (Atomic, Testable, EARS Syntax)
    //
-------------------------------------------------------------------
-------
    package 'Requirements' {
        requirement <REQ_001> 'vacuum safety' {
            doc /* If the vacuum pressure exceeds 1e-5 Torr, then
the system shall disable the Electron Source. */
        }
        requirement <REQ_002> 'spatial resolution' {
            doc /* The system shall provide a spatial resolution of
<= 50 picometers. */
        }
        requirement <REQ_003> 'detector interface' {
            doc /* The Detector shall transmit raw image data to
the Controller at a minimum rate of 1 Gbps. */
        }
    }
```

```
    //
-------------------------------------------------------------------
-------
    // 2) Domain Model (Nouns only; Composition)
    //
-------------------------------------------------------------------
-------
    package 'DomainModel' {
        part def ElectronSource;
        part def HighVoltagePowerSupply;
        part def VacuumChamber;
        part def VacuumPump;
        part def SpecimenStage;
        part def ElectromagneticLens;
        part def Detector;
        part def ImagingSoftware;
        part def UserConsole;

        part def EMSystemDomain {
            part electronSource : ElectronSource;
            part hvPowerSupply : HighVoltagePowerSupply;
            part chamber : VacuumChamber;
            part pump : VacuumPump;
            part stage : SpecimenStage;
            part lenses : ElectromagneticLens;
            part detector : Detector;
            part imagingSoftware : ImagingSoftware;
            part userConsole : UserConsole;
        }
    }

    //
-------------------------------------------------------------------
-------
    // 3) Use Cases (User-Goal Oriented)
```

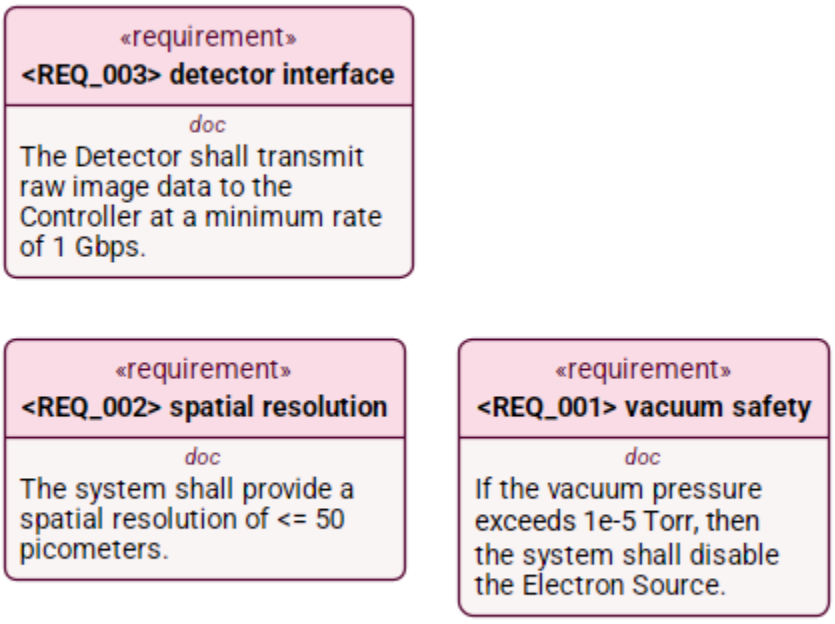
```
    //
-----------------------------------------------------------------------
-------
    package 'UseCases' {
        use case UC_001_Acquire_High_Resolution_Image;
        use case UC_002_Calibrate_Electron_Beam;
        use case UC_003_Prepare_Specimen_Environment;
        use case UC_004_Handle_Vacuum_Leak; // Exception Path
    }

    //
-----------------------------------------------------------------------
-------
    // 4) Subsystems (Object-Oriented Decomposition & State
Behavior)
    //
-----------------------------------------------------------------------
-------
    package 'Subsystems' {
        action def RunVacuumPump;
        action def GenerateBeam;
        action def CaptureInteractions;
        action def ProcessImageData;

        part def IlluminationSubsystem {
            part source : ElectronSource;
            part powerSupply : HighVoltagePowerSupply;
            action generateBeam : GenerateBeam;
        }

        part def VacuumUnit {
            part chamber : VacuumChamber;
            part pump : VacuumPump;
            part stage : SpecimenStage;
            action evacuateChamber : RunVacuumPump;
```

```
            // State Machine for Vacuum Logic
            exhibit state VacuumSM {
                state AtAtmosphere;
                state PumpingDown {
                    do action evacuateChamber;
                }
                state OperationalVacuum;
                transition first AtAtmosphere accept startPumping
then PumpingDown;
                transition if PressureAchieved from PumpingDown
then OperationalVacuum;
            }
        }

        part def ElectronColumn {
            part lenses : ElectromagneticLens;
            part detector : Detector;
            action capture : CaptureInteractions;
        }

        part def ControllerSubsystem {
            part console : UserConsole;
            part software : ImagingSoftware;
            action processData : ProcessImageData;
        }

        // Top-Level System Composition
        part def ElectronMicroscope {
            part illumination : IlluminationSubsystem;
            part vacuum : VacuumUnit;
            part column : ElectronColumn;
            part controller : ControllerSubsystem;
        }
    }
}
```

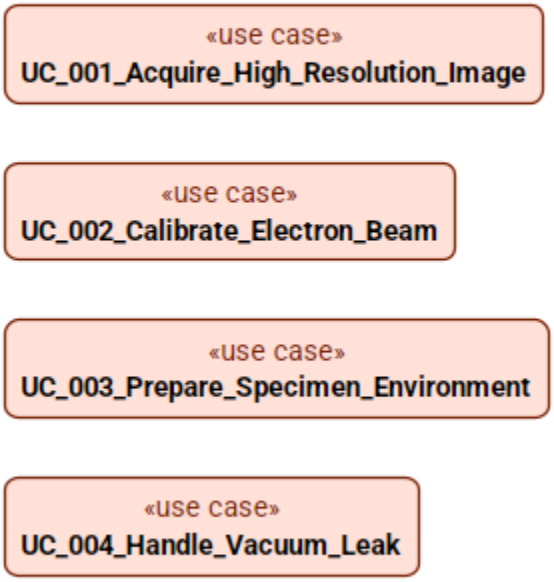
Domain Diagram



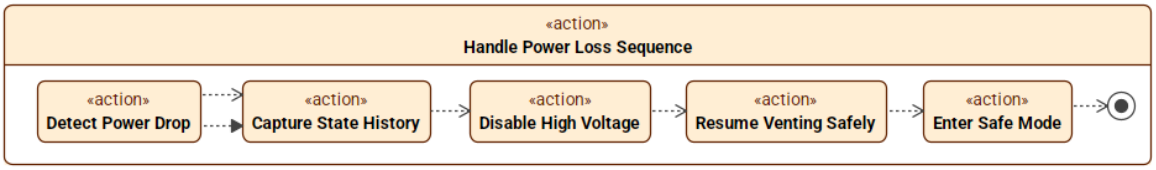
Requirements Diagram



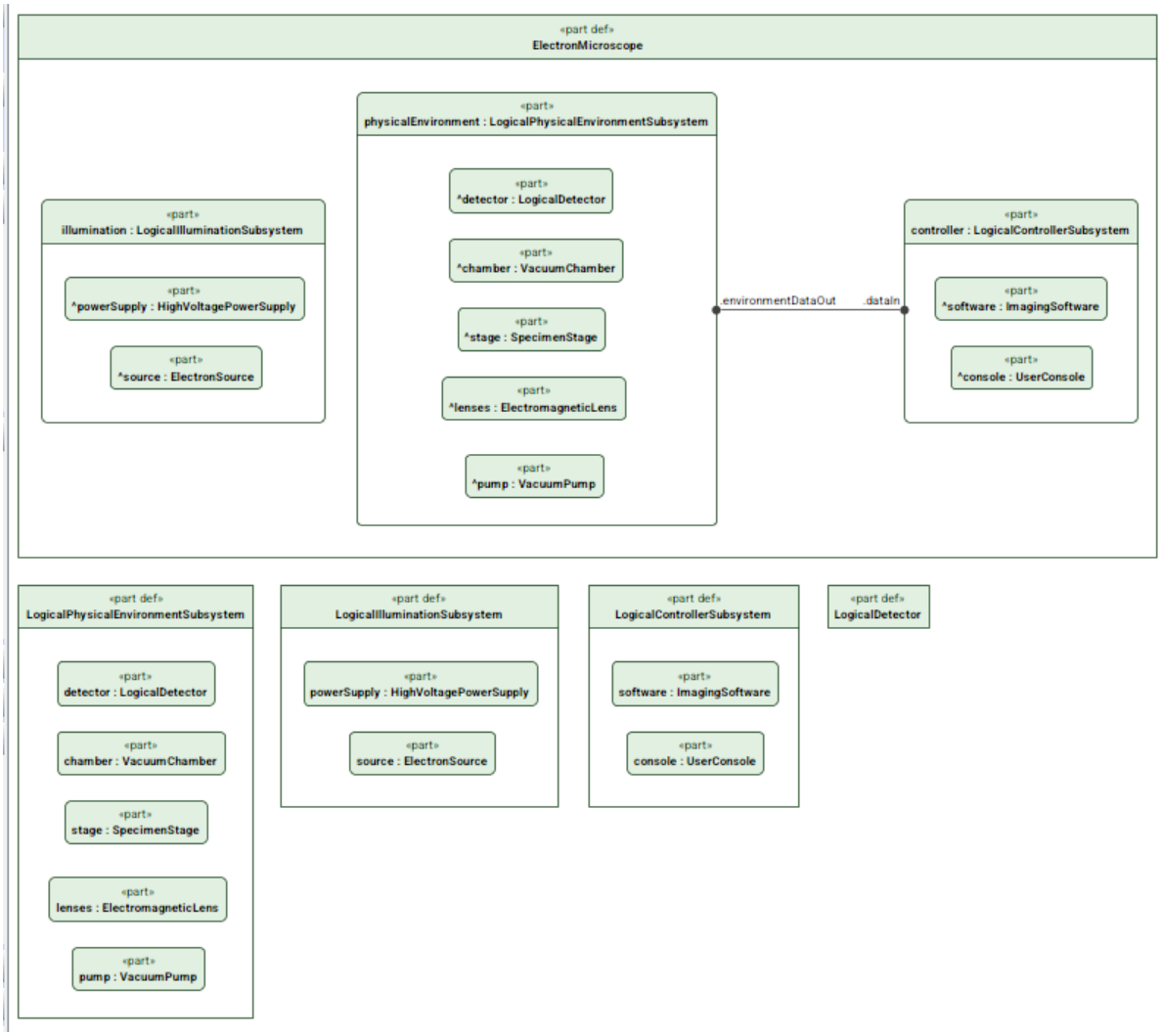
Use Case Diagram



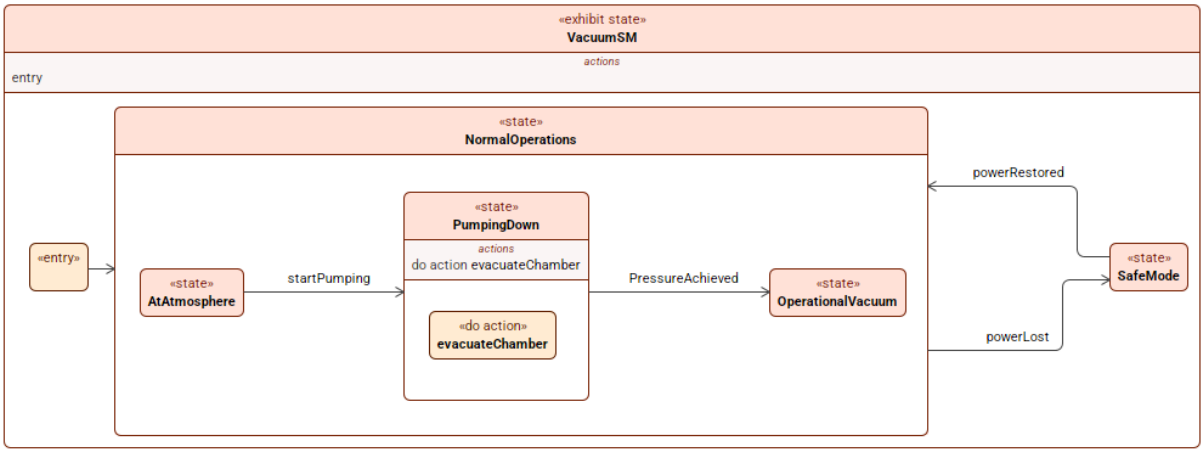
Action-Activity Diagram



Future-State Block Diagram



State Diagram



Phase 9: Graphical Representation (Cameo / CSM Output)

While the underlying model acts as the single source of truth, standard SysML views are generated to facilitate communication with stakeholders, such as the Board Meeting. The graphical artifacts natively produced by the modeling tool include:

- **System Package Diagram (`pkg`):** Visualizes the namespace containment and dependencies of the `Electron Microscope System Model` into its four core packages (`Requirements`, `DomainModel`, `UseCases`, `Subsystems`).
    - *Graphical Rendering:* [PLACEHOLDER] - *The actual exported graphic from the CSM environment is undefined here.*
- **Requirements Diagram (`req`):** Displays the functional, performance, and interface requirements along with their `«satisfy»` relationships to logical subsystem blocks.
    - *Graphical Rendering:* [PLACEHOLDER] - *The visual block layout mapping REQ_001, REQ_002, and REQ_003 is undefined.*
- **Use Case Diagram (`uc`):** Outlines the system boundary (the Electron Microscope), external actors (Lab Technicians, Regulators), and the identified Enterprise Use Cases. It specifically visualizes the exception paths (e.g., `Handle Vacuum Leak`) branching from the primary goals.
    - *Graphical Rendering:* [PLACEHOLDER] - *The drawn ellipses and stick figures mapping the actor associations are undefined.*
- **To-Be Domain & Logical Architecture Block Definition Diagrams (`bdd`):** Displays the classification tree and structural components. The To-Be BDD focuses on the `EMSystemDomain` and its reference associations to actors. The System Logical Architecture BDD displays the `ElectronMicroscope` block and its composite associations (diamond-tipped lines) to the four primary subsystems.
    - *Graphical Rendering:* [PLACEHOLDER] - *The rendered SysML blocks and composition connectors are undefined.*
- **State Machine Diagram (`stm`):** Visually renders the dynamic behavior of the `VacuumUnit` subsystem. It depicts the simple states (`AtAtmosphere`, `PumpingDown`, `OperationalVacuum`), the `Do` behaviors, and the explicit event triggers (e.g., `PressureAchieved`) driving the transitions.
    - *Graphical Rendering:* [PLACEHOLDER] - *The exact visual mapping of the round-cornered state rectangles and transition arrows is undefined.*

Phase 6: Synthesize Candidate Physical Architectures

Within the OOSEM framework, defining the physical architecture entails answering *how* the logical components will be physically realized. However, the Guardrail AI framework explicitly guided the conceptual design session to focus on a "Domain-Driven Logical Architecture" to prevent premature technical obsession and "over-modeling".

- **Hardware/Software Allocation:** [PLACEHOLDER] - *The specific allocation of logical subsystems (e.g., `IlluminationSubsystem`, `VacuumUnit`) to physical hardware nodes or specific software deployment environments is undefined.*
- **Vendor Constraints:** The team explicitly avoided modeling the internal circuitry of the vacuum pump because it is a vendor-supplied part.
- **Physical Interfaces (Ports):** [PLACEHOLDER] - *The specific physical ports, pin-outs, and cabling (e.g., standard vs. full/proxy ports) connecting the High Voltage Power Supply to the Electron Source are undefined.*

Phase 7: Optimize and Evaluate Alternatives

This stage involves building parametric models to conduct trade studies and evaluate candidate architectures against measures of effectiveness.

- **Parametric Models (Mathematical Constraints):** [PLACEHOLDER] - *The parametric diagrams evaluating alternative configurations for cost, mass, power throughput, or vacuum pump-down times are undefined.*
- **Trade Studies:** [PLACEHOLDER] - *Evaluations of alternative electron source technologies (e.g., thermionic vs. field emission) or vacuum pump technologies (e.g., turbomolecular vs. ion pumps) are undefined.*

Phase 8: Behavior Verification & "Rainy-Day" Scenarios

While the primary "Happy Path" (`Acquire High-Resolution Image`) and a critical exception path (`Handle Vacuum Leak`) were defined, robust system architectures require extensive stress-testing against edge cases and system failures.

- **Power Loss/Safe Mode Sequence:** The script identifies the need for a `Safe Mode` utilizing a History Pseudostate to remember the system's state and resume venting safely if power is lost.
    - *Detailed Safe Mode Activity/Sequence Diagrams:* [PLACEHOLDER] - *The specific actions, control flows, and signal event triggers for this power-loss sequence are undefined.*

- **Additional Rainy-Day Scenarios:** The conceptual session ends with an explicit prompt to generate a specific list of additional "Rainy-Day" scenarios to further stress-test the architecture.
    - *System Fault Scenarios:* [PLACEHOLDER] - *Additional exception use cases (e.g., detector failure, software crash, high-voltage arcing) are undefined.*

Conclusion of the Conceptual Phase

This OOSEM conceptual report successfully bridges the "Acquisition-Development Gap." By shifting from static text to an executable SysML v2 representation, the project management and systems engineering teams have established a verifiable logical architecture. Every model element—from the 50-picometer spatial resolution requirement to the Vacuum Unit state machine—traces directly back to the researchers' core concerns of image quality, lab efficiency, and safety.

This concludes the comprehensive OOSEM report based on your provided conceptual design session script.