Based on the domain information provided for the GPCA system and following the process of instantiating the safety case pattern, I will replace the placeholders with the system-specific information. I will also remove the additional decorators and aim to produce a hierarchical tree format representation using dashes (-) to denote different levels.

Here is the instantiated safety case in a hierarchical tree format:

```

- G1: Operational safety is verified in GPCA system

- C1: All specs are given by spec.allInstances()

- S1: Argument over the satisfaction of all specs over the GPCA system

- S2: Argument over reliability in all suitable levels of GPCA system (undeveloped)

- G2: All operational hazards are mitigated

- C2: Operational hazards are given by operational\_hazard.allInstances()

- S3: Argument over operational hazards

- G3: Overinfusion is mitigated

- C3: All related scenarios are given by Overinfusion

- S4: Argument over the applied scenarios of Overinfusion

- G4: Overinfusion is mitigated under scenario 1

- C4: All related specs are given by scenario.spec.allInstances()

- S5: Argument over all specs related to scenario 1

- G5: Spec SR1.1 is appropriate for scenario 1

- C5: Properties are given by spec.property.allInstances()

- S6: Argument the appropriateness of SR1.1 over properties

- G6: Flow rate sensor is appropriate for SR1.1

- C6: Source is given by property.source

- S7: Argument over the source of the Flow rate sensor definition

- G7: Flow rate sensor source is appropriate and trustworthy

- Sn1: FDA standard

- G8: Flow rate sensor definition is sufficient

- Sn2: Flow rate sensor

- G3: Underinfusion is mitigated

- C3: All related scenarios are given by Underinfusion

- S4: Argument over the applied scenarios of Underinfusion (undeveloped)

- G4: Underinfusion is mitigated under scenario 2

- C4: All related specs are given by scenario.spec.allInstances()

- S5: Argument over all specs related to scenario 2 (undeveloped)

- G5: Spec SR1.2 is appropriate for scenario 2

- C5: Properties are given by spec.property.allInstances()

- S6: Argument the appropriateness of SR1.2 over properties (undeveloped)

- G6: Period (to trigger the underinfusion alarm) is appropriate for SR1.2

- C6: Source is given by property.source

- S7: Argument over the source of the Period (to trigger the underinfusion alarm) definition

- G7: Source of period definition source is appropriate and trustworthy

- Sn1: Expertise opinion

- G8: Definition of period definition is sufficient

- Sn2: Period (to trigger the underinfusion alarm)

- G3: Dose limit exceeded due to excessive bolus requests is mitigated (undeveloped)

- C3: All related scenarios are given by Dose limit exceeded

- S4: Argument over the applied scenarios of Dose limit exceeded (undeveloped)

- G3: Bolus volume/concentration too high is mitigated (undeveloped)

- C3: All related scenarios are given by bolus volume/concentration too high

- S4: Argument over the applied scenarios of bolus volume/concentration too high (undeveloped)

```

In this instantiated safety case:

- I’ve replaced placeholders with the GPCA system-specific information, such as the different operational hazards like "Overinfusion" and "Underinfusion."

- The pattern decorators and placeholders have been removed.

- The hierarchical tree structure clarifies the argument flow from operational safety verification to the lowest-level detailed arguments and evidence supporting the claim.

Several subgoals and strategies remain undeveloped as denoted (e.g., "Argument over the reliability in all suitable levels of GPCA system"), indicating areas requiring further development in the assurance case.