| Requirement ID | Requirement Description | UML reference | Implementation module | Test case id |
|---|---|---|---|---|
| R1 | Input field for entering waypoints (latitude, longitude, altitude). | enterWaypoint method, waypoints and waypointInputField attributes in FlightPlanManagement class in UML class diagram. | Ui module | N/A |
| R2 | Input fields for speed restrictions and expected times of arrival at each waypoint. | enterSpeedRestriction and enterArrivalTime method and speedInputField and timeInputField attributes in FlightPlanManagement class in UML class diagram | Ui module | N/A |
| R3 | A submit button to load and activate the flight plan. | loadFlightPlan method and submitButton attribute in FlightPlanManagement class in UML class diagram | Ui module | N/A |
| R4 | Visual display (map) showing current position, planned route, and waypoints. | displayMap method in FlightPlanManagement class in UML class diagram | Ui module | N/A |
| R5 | Engage autopilot when at cruising altitude, and disengage it for landing. | engageAutoPilot and disengageAutoPilot method in AutoPilotControlPanel class in UML class diagram | Ui module | N/A |
| R6 | System should provide a easy way for pilots to manually override autopilot. | disengageAutoPilot method in AutoPilotControlPanel class in UML class diagram | UI module | N/A |

| | | | | |
|---|---|---|---|---|
| R7 | Respond quickly and appropriately to any hazards. | triggerAudibleAlert method and visualAlert method in HazardAlertsDisplay | UI module | N/A |
| R8 | Indicator lights for autopilot status (engaged, disengaged, fault condition). | displayIndicatorLights method and autoPilotStatus attribute in AutoPilotControlPanel class in UML class diagram | UI module | N/A |
| R9 | Controls for manual override: altitude adjustment, speed, heading. | adjustAltitude, adjustSpeed, adjustHeading method in AutoPilotControlPanel class in UML class diagram | UI module | N/A |
| R10 | Digital readouts for airspeed, altitude, pitch, roll, yaw, and engine parameters. | displayAirspeed, displayAltitude,displayPitch, displayRoll, displayYaw, displayEngineParameters method and airspeed, altitude,pitch,roll,yaw and engineParams attributes in SensorDataDisplay class in UML class diagram | UI module | N/A |
| R11 | Visual indicators for data update frequency (e.g., colour change or blinking to indicate fresh data). | updateAirspeed, updateAltitude, updatePitch, updateRoll,updateYaw, updateEngineParameters,displayIndicator, colorChange method in SensorDataDisplay class in UML class diagram | UI module | N/A |
| R12 | Interface should be responsive to pilot input and button presses. | engageAutoPilot and disengageAutoPilot, with engageAutoPilotButton field. | UI module | N/A |

| | | | | |
|---|---|---|---|---|
| R13 | The UI should have a dedicated section for hazard warnings and mitigation actions – these actions must be providing as soon as possible after the initial hazard warning has been raised – ideally concurrently . | Hazard alert class in UML and issueHazardAlert method | UI module | N/A |
| R14 | Interface should include a checklist or action plan for emergency procedures. | displayCheckList method in HazardAlertsDisplay class. | UI module | N/A |
| R15 | Audible and visual alerts should be provided for immediate hazards. | triggerAudibleAlert and visualAlert method in HazardAlertsDisplay class. | UI module | N/A |
| R16 | Attitude Sensor should measure pitch (nose up/down), roll (wing up/down), and yaw (nose left/right). | AttitudeSensor class in UML with the getters and setter methods within the class. | Sensor | |
| R17 | The data format for | scheduleAttitudeSensorUpdate method in Coresystem. | Core | |

| | | | | |
|---|---|---|---|---|
| | orientation data should be of degrees from the horizon for pitch and roll and magnetic heading for yaw with an update frequency of every 500 milliseconds . | | | |
| R18 | Operational ranges of orientation data: Pitch: -30° to 30°, Roll: -60° to 60°, Yaw: -180° to 180°. Identify and manage exceedances through system alerts and corrective actions. | detectFaults in CoreSystem, as well as getMinPitch, getMaxPitch, getMinYaw, getMaxYaw, getMinRoll, getMaxRoll methods in AttitudeSensor class. | Core and sensor | |
| R19 | Altitude Sensor should provide barometric and gps data. | Altitudesensor class in UML with the getters and setter methods of that class. | Sensor | |
| R20 | Altitude sensor data should be in the format of AMSL with update frequency of every 500 | scheduleAltitudeSensorUpdate method in Coresystem. | CoreSystem | |

| | | | | |
|---|---|---|---|---|
| | milliseconds. | | | |
| R21 | Altitude sensor should be capable of measuring altitude within -1000 to 50000 feet AMSL. | detectFaults in CoreSystem, as well as getMinAltitude and getMaxAltitude in AltitudeSensor class. | Core, Sensor | |
| R22 | The airspeed sensor should be able represent the aircraft's speed relative to the surrounding air and be in knots. | getCurrentAirspeed, setCurrentAirspeed method and currentAirspeed attribute in airspeed sensor class in UML. | Sensor | |
| R23 | The airspeed sensor should also be able to represent aircraft speed between 50 and 500 knots and have update frequency of every second. | scheduleAirspeedSensorUpdate method in AirspeedSensor class, as well as getMaxSpeed and getMinSpeed methods and detectFault method in Coresystem class. | Sensor, Core | |
| R24 | Engine parameters should measure engine thrust and fuel flow in data format of thrust in pounds- | setFuelFlow, getFuelFlow, getCurrentThrust, setCurrentThrust method and updatefrequency attribute in Engine class UML. | Sensor | |

| | | | | |
|---|---|---|---|---|
| | force (lbf) and should have update frequency of every second. | | | |
| R25 | The system must prioritize safety with redundancy included for sensors. | 3 to 1 relationship from the sensors class to sensordatadisplay and sensordata class in UML, alongside ids for each sensor. As well as with core system and sensor class. | Sensor, core | |
| R26 | The system shall handle sensor errors and unexpected events without crashing. | detectFault method in Coresystem class for UML. | Core | |
| R27 | Hazard Mitigation Strategies: Implement fault detection, fault tolerance, and fail-safe mechanisms to handle abnormal sensor values, ensuring the system can safely manage potential hazards without catastrophic failures. | detectFaults,handleSensorFault, notifyPilot methods in coresystem class. getMinSpeed, getMaxSpeed, getMinAltitude, getMaxAltitude, getMinYaw, getMaxYaw, getMinRoll, getMaxRoll, getMinPitch, getMaxPitch from the airspeed, altitude, attitude classes respectively. | Core and Sensors module | |
| R28 | Sensors including airspeed sensor, | 3 to 1 relationship from the sensors class to sensordatadisplay and sensordata class in UML. As well as with core system and sensor class. | Sensor, core | |

| | | | | |
|---|---|---|---|---|
| | altitude sensor, and attitude sensor should operate in a 2oo3 architecture, providing backup in the case of a sensor failing. | | | |
| R29 | Execution Check Parameters: After sending a control signal, the system must verify the execution by reading back the relevant sensor data. | Sendcontrolsignals method in autopilotsystem. sendSensorData and updateFromSensorData method for sensors and engine parameters. | Autopilot, Sensor | ID4 , ID5 |
| R30 | For the Autopilot Control Frequency, the system should send control signals to the aircraft's control surfaces (elevators, ailerons, rudders) and engine control systems. | Sendcontrolsignals method in autopilotsystem. | Autopilot, Sensor | ID4 |

| | | | | |
|---|---|---|---|---|
| R31 | Success Criteria: A control signal is considered successfully executed if the sensor data reflects the expected change within a margin of error of ±2% for the control surfaces and ±5% for engine parameters, within 1 second of command issuance. | verifyExecution method in autopilotsystem class. | Autopilot, sensor | ID4 , ID5 |
| R32 | Failure Handling: If the execution check fails after sending a control signal, the system must attempt to resend the command up to three times before alerting the pilot to the issue via the user interface. | verifyExecution, resendControlSignal, and alertUserInterface method in autopilotsystem class. | Autopilot, Sensor | ID6 |
| R33 | The system should dynamically adjust | setCurrentThrust(double newThrust) in engine class | Sensor | |

| | | | | |
|---|---|---|---|---|
| | engine thrust proportional to airspeed. | | | |
| R34 | System should accurately model the effects of thrust adjustments on altitude changes | setCurrentThrust(double newThrust) in engine class | Sensor | |
| R35 | System should model how changes in thrust affect the aircraft's pitch, and in scenarios of asymmetric thrust, its roll and yaw. | setCurrentThrust(double newThrust) in engine class | Sensor | |
| R36 | The maximum thrust will be 374 kN per engine. | MAX_THRUST attribute in engine | Sensor | |
| R37 | The minimum thrust will be 10kN per engine. | MIN_THRUST attribute in engine | Sensor | |
| R38 | Integration of autopilot system with sensors to gather data. | sendControlSignal in autopilotsystem, executeControlSignal, updateFromSensorData, and sendSensorData methods in Control surface. | Autopilot, sensor | ID4, ID5, ID6 |
| R39 | Autopilot system should incorporate redundancy in the case of software bugs. | 1 to many relationship between AutoPilotControlPanel and AutoPilotSystem. Alongside checkerrorinautopilot method in core system and checkActive, errorInAutoPilot, and healthCheck method in autopilotsystem | Core and Autopilot module | ID7 |

| | | | | |
|---|---|---|---|---|
| R40 | The system shall ensure continuous and reliable communication between the autopilot system and critical components, such as the engine control system, to prevent loss of control over throttle settings or other vital parameters. | The relationship of Autopilotsystem engine and control system helps ensure the reliable communication alongside methods like sendControlSignals and sendSensorData. | Autopilot, sensor module | ID4, ID5, ID6 |