Unmanned Aerial Vehicles Sensing
Missions Specification using SensorML

Bachelor’s Thesis in Computer Engineering

CEYLAN AVCI
DETAILS
First Name, Surname: Ceylan Avci
University: Halmstad University, Sweden
Degree Program: Computer Engineering
Title of Thesis: Unmanned Aerial Vehicles Sensing Missions Specification using SensorML
Academic Supervisor: Edison Pignaton de Freitas
ABSTRACT

SensorML is a programming language which an user can write any type of sensor specification and group of sensors as a database in XML form. This thesis is written with the target of showing SensorML specifications and clarifying it with an example of type of UAV and its camera sensors. Four type of UAV platforms are used as examples:

- a helicopter
- a blimp
- a quad copter
- and a aircraft

and five types of camera sensors are given as examples of sensors that equip the UAV platforms:

- low light
- HD camera
- thermal camera
- and micro analog camera

These UAV platforms and sensors are combined to represent four different UAV systems in SensorML.

By the end this database information will be used in a small application to show SensorML programming validation.

The usage of this utilization of the SensoML is to support the selection of the best fit in relation to UAV platform and sensor device for a given mission specified using the language. The main goal of the thesis is to highlight this utility.
ACKNOWLEDGEMENT

To be honest I am grateful to everybody who teaches me even “one word” from the first moment to this moment in my education life. Learning never finish so I am eager to see what will be after that. This word really not comes for obligatory I am really fell happiness with every moment in Halmstad University and Sweden. My special thanks for my supervisor Edison Pignaton de Freitas. He is really great teacher who supported me during my bachelor thesis and gives really vital advices which succeed to do my work. It would be impossible for me to finish my thesis without his help. So I want to thanks him twice.

Also I would like to thank Kenneth Nilsson. He helped me to clarify my way during writing my thesis with his important suggestions. At the moment of confusing my mind he is a good problem solver. Thanks to him.

To the end I want to represent my deepest regards to my lovely family which not supported me only doing my thesis also in my every moment. I want to put a little love in my regards and thanks to them countless times.
# TABLE OF CONTEXTS

<table>
<thead>
<tr>
<th>Details</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>III</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENT</td>
<td>IV</td>
</tr>
<tr>
<td>TABLE OF CONTEXTS</td>
<td>V</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>VII</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>VIII</td>
</tr>
<tr>
<td>LIST OF LISTINGS</td>
<td>IX</td>
</tr>
</tbody>
</table>

1) **INTRODUCTION** ........................................................................................................... 1
   1.1. **BACKGROUND** ........................................................................................................ 1
   1.2. **PROJECT OBJECTIVE** ............................................................................................ 1
   1.3. **PROJECT OUTLINE** .................................................................................................. 2

2) **METHODS** ....................................................................................................................... 3
   2.1. **SENSOR MODEL LANGUAGE (SENSORML) [17]** ............................................................... 3
   2.2. **SENSORML CONCEPTUAL MODELS** ................................................................................ 3
       2.2.1. **PROCESSTYPE** ................................................................................................... 4
       2.2.2. **PROCESSMODEL** .................................................................................................. 5
       2.2.3. **PROCESS METHOD** ............................................................................................... 6
       2.2.4. **PROCESSCHAIN** ................................................................................................... 6
   2.3. **NON-PHYSICAL PROCESSES** ....................................................................................... 6
   2.4. **PHYSICAL PROCESSES** .............................................................................................. 7
       2.4.1. **COMPONENT** ....................................................................................................... 8
       2.4.2. **SYSTEM** ............................................................................................................. 8
   2.5. **PROCESS METADATA GROUP** ..................................................................................... 8

2.6. **SENSORS** ..................................................................................................................... 10
   2.6.1. **DIGITAL SENSORS** ................................................................................................. 10
   2.6.2. **ANALOG SENSORS** ................................................................................................ 11

2.7. **CLASSIFICATION OF SENSORS [16]** ........................................................................... 11
   2.7.1. **ACCELERATION, SHOCK AND VIBRATION SENSORS** ................................................... 11
   2.7.2. **BIOSENSORS** ......................................................................................................... 12
   2.7.3. **FLOW AND LEVEL SENSORS** ................................................................................. 13
       2.7.3.1. **THERMAL ANEMOMETERS** ............................................................................... 14
       2.7.3.2. **DIFFERENTIAL PRESSURE MEASUREMENT** ................................................... 14

3) **UNMANNED AERIAL VEHICLES SENSING MISSIONS SPECIFICATION USING SENSORML** ...... 15
   3.1. **UNMANNED AERIAL VEHICLES** ................................................................................ 15
       3.1.1. **DRAGANFLYERX6** .............................................................................................. 15
       3.1.2. **DRAGANFLYERX6 HELICOPTER TECH SPECS [2]** ............................................. 16


<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.3. DRAGANFLYERX8</td>
<td>17</td>
</tr>
<tr>
<td>3.1.5. DRAGANFLY TANGO UAV AEROPLANE</td>
<td>19</td>
</tr>
<tr>
<td>3.1.6. DRAGANFLY TANGO AEROPLANE TECHSPECS [6]</td>
<td>20</td>
</tr>
<tr>
<td>3.1.7. CIRRUS 1000 BLIMP</td>
<td>21</td>
</tr>
<tr>
<td>3.1.8. CIRRUS 1000 BUMPTECH SPECS [8]</td>
<td>21</td>
</tr>
<tr>
<td>3.2. CAMERAS</td>
<td>22</td>
</tr>
<tr>
<td>3.2.1. IP VIDEO CAMERA</td>
<td>22</td>
</tr>
<tr>
<td>3.2.2. DIGITAL STILL CAMERA WITH 720P VIDEO RECORDING</td>
<td>23</td>
</tr>
<tr>
<td>3.2.3. LOW LIGHT CAMERA</td>
<td>23</td>
</tr>
<tr>
<td>3.2.4. THERMAL INFRARED CAMERA</td>
<td>24</td>
</tr>
<tr>
<td>3.2.5. MICRO ANALOG VIDEO CAMERA</td>
<td>24</td>
</tr>
<tr>
<td>3.2.6. VIDEO&amp; STILL CAMERA SYSTEM SPECIFICATIONS [14]</td>
<td>25</td>
</tr>
<tr>
<td>3.3. IMPLEMENTATION</td>
<td>27</td>
</tr>
<tr>
<td>3.3.1. DATABASE</td>
<td>29</td>
</tr>
<tr>
<td>3.3.1.1. KEYWORDS</td>
<td>30</td>
</tr>
<tr>
<td>3.3.1.2. IDENTIFICATION</td>
<td>30</td>
</tr>
<tr>
<td>3.3.1.3. CLASSIFICATION</td>
<td>31</td>
</tr>
<tr>
<td>3.3.1.4. CHARACTERISTICS</td>
<td>32</td>
</tr>
<tr>
<td>3.3.1.5. CAPABILITIES</td>
<td>34</td>
</tr>
<tr>
<td>3.3.1.6. CONNECTION LIST</td>
<td>36</td>
</tr>
<tr>
<td>3.4. SENSING MISSION MANAGEMENT PROGRAM</td>
<td>37</td>
</tr>
<tr>
<td>4) RESULTS</td>
<td>43</td>
</tr>
<tr>
<td>5) CONCLUSION</td>
<td>44</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>45</td>
</tr>
<tr>
<td>APPENDIX</td>
<td>47</td>
</tr>
<tr>
<td>UAV XML DATABASES</td>
<td>47</td>
</tr>
<tr>
<td>1.DRAGANFLYERX6</td>
<td>47</td>
</tr>
<tr>
<td>2.DRAGANFLYERX8</td>
<td>53</td>
</tr>
<tr>
<td>3.DRAGANFLY TANGO</td>
<td>59</td>
</tr>
<tr>
<td>4.CIRRUS 1000 BLIMP</td>
<td>65</td>
</tr>
<tr>
<td>CAMERA XML DATABASES</td>
<td>71</td>
</tr>
<tr>
<td>1.MICRO ANALOG</td>
<td>71</td>
</tr>
<tr>
<td>2.THERMAL INFRARED</td>
<td>80</td>
</tr>
<tr>
<td>3.LOW LIGHT</td>
<td>92</td>
</tr>
<tr>
<td>4.HD CAMERA</td>
<td>100</td>
</tr>
<tr>
<td>CODE FOR UAV QUESTIONNAIRE</td>
<td>110</td>
</tr>
<tr>
<td>CODE CAMERA UAV QUESTIONNAIRE</td>
<td>116</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 1: Conceptual model for Process .................................................................3
Figure 2: Alternative model view for physical and non physical processes ............5
Figure 3: ProcessMetadataGroup Properties .........................................................8
Figure 4: Graphical vision of discrete and analog Signal ......................................10
Figure 5: (a) Switch (b) Sharp IR detector .............................................................10
Figure 6: (a) Potentiometer Sensor (b) CdS cell wiring diagram ...........................11
Figure 7: Basic piezoelectric accelerometer construction ....................................11
Figure 8: Requirements for an implantable glucose sensor ...................................12
Figure 9: Bioreceptor Configuration .....................................................................13
Figure 10: Pitot tube ...............................................................................................14
Figure 11: DraganflyerX6 helicopter ....................................................................15
Figure 12: DraganflyerX6 Helicopter specifications ............................................16
Figure 13: DraganflyerX8 quadcopter ..................................................................17
Figure 14: DraganflyerX8 quadcopter specifications ..........................................18
Figure 15: Draganfly Tango Aero plane ...............................................................19
Figure 16: Draganfly Tango Aero plane specifications .........................................20
Figure 18: IP Video Camera ..................................................................................22
Figure 19: Digital Still Camera with 720p Video Recording .................................23
Figure 20: Low light Camera ................................................................................23
Figure 21: Thermal infrared Camera ......................................................................24
Figure 22: Micro Analog Video Camera ...............................................................24
Figure 23: System 1 and 2- Cameras with Quad copter and Cameras with Blimp ...28
Figure 24: System 3 and System 4-Cameras with aero plane and cameras with helicopter ...28
Figure 26: Web page about UAV Questions .......................................................38
Figure 27: Web page about Camera Questions ....................................................42
Figure 28: Result pages .........................................................................................42
LIST OF TABLES
Table 1: OSI Layers for SensorML..........................................................7
Table 2: General using places of Metadata Group Properties .........................9
Table 3: DraganflyerX6 Specifications ........................................................16
Table 4: DraganflyerX8 Specifications ........................................................18
Table 5: Draganfly Tango Aero plane specifications ......................................20
Table 6: Cirrus 1000 Blimp Tech Specs .......................................................22
Table 7: Video & Still Camera System Specifications .....................................25
LIST OF LISTINGS
Listing 3.1: Keywords (SensorML Code) .................................................................30
Listing 3.2: Identification (SensorML Code) ..........................................................31
Listing 3.3: Classification (SensorML Code) .........................................................32
Listing 3.4: Characteristics (SensorML Code) .....................................................34
Listing 3.5: Capabilities (SensorML Code) .........................................................36
Listing 3.6: Contact List (SensorML Code) ..........................................................37
Listing 3.7: An example of an quantity part (SensorML Code) ...............................39
Listing 3.8: UAV Questionaire Code (Just for first question as given example) .........41
Listing 3.9: Camera Questionaire Code (differences from UAV questionnaire) ...........41
1) INTRODUCTION

1.1. BACKGROUND
Humans always have made research about doing works easily with less time and less human power. If the subject is about lifting an object or transporting it from one place to the other it can be handled with basic actuator systems. But if job that has to be done is about defining the object, find its temperature, measuring size or color actuator systems are unable to do that kind of task. Sensors play role at that point.

Increasing technology also increased the available types and number of sensors in condition of its applicable areas. Humans want to make all kind of jobs in easiest way and that created automation. Automation cannot be done without sensors because of detecting abilities. The desire of using automation in everywhere also expanded and made confusion of sensor classification. SensorML is a programming language that classifies sensors and observations as a database in XML form.

SensorML created by OGC systems in the target of classifying physical and non physical events. Non physical event can be such an equation etc. Physical event can be a sensor, detector, a device or a machine. SensorML saves data such as sensor type, what it does, where it is used, its size, color, relations with surrounding environment and its own specialties, its weight that help us to define and recognize the sensor. Everything can be considered as a system in SensorML under the condition of being related with other sensors or not. In other words, one sensor constitutes a system but combination of more than one sensor also can constitute a system. For instance a thermal camera must contain a detector that sense environment temperature and also another type of sensor that analyses surrounding objects visions. That means thermal camera built up more than two sensors or detector relativity. But as another example such in this thesis an UAV and several cameras inside it also totally constitute a system which is actually not-related to each others.

1.2. PROJECT OBJECTIVE
The aim of thesis to show the use of SensorML to specify sensing missions in which mobile sensor nodes, (Unmanned Aerial Vehicles - UAVs) are used. Using SensorML it is possible to classify several UAVs (helicopter, quadcopter, blimp and aero plane) and cameras (micro analog, HD camera, lowlight and thermal camera) datasheet specifications as a database in XML format.
A program is created with aim of using databases to choose most appropriate UAV and camera in a specific condition of a mission. The program show how the language is used to define the missions and how this specification is then used to select the most appropriate UAV-sensor combination.

1.3. PROJECT OUTLINE
This thesis made up following parts:
• Chapter 2: The methods about writing thesis General specialties of Sensors and types of them; Information about SensorML.
• Chapter 3: Implementation of thesis- Information about Chosen UAVs and Cameras and their datasheet specialties. 2 Steps for succeed thesis first creating databases with using SensorML second, writing code with the aim of find best option in a given test condition.
• Chapter 4: Result of thesis
• Chapter 5: Conclusion.
2) METHODS

2.1. SENSOR MODEL LANGUAGE (SENSORML) [17]
SensorML is a programming language which implements the specifications of sensors and observations in XML format.

2.2. SENSORML CONCEPTUAL MODELS
Transducers, actuators and processors normally consist of process component, sensors and platforms normally consist of systems in SensorML. All these elements stuff can be modeled as a component of system, thus everything is assumed as a process in SensorML. Gathering processes including data about components, parameters as input consists of process chains.

SensorML can be classified as a process which describes the language as BPEL or MATLAB Simulink. Furthermore, it is possible to convert a project which is carried out with BPEL or MATLAB Simulink to SensorML. Unified Model Language (UML) is used in SensorML for creating models. These models are conceptual and it is unsuitable write XML Schema automatically.

![Figure 1: Conceptual model for Process](image)

Process chains are also linkable, which means adding this code into an observation or the observation linage is also possible. Given inputs with required parameters will produce
outputs and this makes a collection of physical and non-physical processes executable in SensorML. All of these relations between process, process chain and system are shown in Figure 1.

2.2.1. PROCESSTYPE
Abstract Feature forms the minimum part of process. Basically, a feature has a name and description. Additionally, Abstract Process is derived from abstract feature and it has three properties such as inputs, outputs and parameters called AnyData. SWE Framework supports linkage between SWE components, encodings and services and this provides consistency in SensorML.

Inputs and outputs can be defined as ports in SensorML when a process communicates with other process. The type of input does not change it because value like yes or no will also be a value that is a number.

It is available to write all properties in the format of inputs, outputs and parameters. Besides more understandable codes, more qualified results and corrective in important resource discoveries, SensorML provides additional data called metadata group. Copying or linking one metadata group to another metadata group is possible.

Atomic process consists of one process. Composite process which is called as composite properties group consists of process chain, system processes and connections.
Figure 2: Alternative model view for physical and non-physical processes

Figure 2 shows classification of processes. It is possible to separate the processes as physical and non-physical processes. Physical process is different from non-physical process and its location is important, therefore physical processes can also take properties including spatial reference frame, temporal reference frame, bounded by, position, interface and these properties consist of physical properties group.

2.2.2. PROCESSMODEL

Process chains consist of pure an atomic process which is called as a process model. Process class gives all of its properties to process model and the process model which provides description consist of the property of method.
2.2.3. PROCESS METHOD
Process Method converts inputs to outputs with the help of parameters. The aim of SensorML is to create executable processes where a prior knowledge which makes it rather complex is needed. The data communication is supposed to be understood by SensorML with the help of process method in both inside of process chains and outside of processes.

There is a list of metadata for ProcessMethod;

- IOStructure Definition: Defines required and optional input, output and the types of parameter.
- Algorithm Definition: Unambiguously specifies the algorithm that can be implemented (in software) or enabled (in hardware) for this process.
- Implementation Definition: Provides information regarding executable implementations for this process.

2.2.4. PROCESSCHAIN
Process chains consist of processes which have input and these processes have provided parameters in order to gain the required result. Only a few processes produce outputs and these outputs will be another processes input. This leads process chain to have its own inputs, outputs and parameters.

In SensorML, everything is assumed as a process, thus it is possible to assume a process chain like a process which can consist of a chain in another process chains with the specialty of being Composite Design pattern.

2.3. NON-PHYSICAL PROCESSES
If the main object used in SensorML is a detector, actuator or sensor, it is also important to specify its location and the specialties of interfaces. In other words, this makes the process
physically a touchable object. Consequently, if the process makes up from mathematically operation, it consists of non-physical process.

2.4. PHYSICAL PROCESSES
Devices such as transducers (detectors and actuators), sensor systems, samplers, and sensor platforms must be defined with mathematical processes, but additionally, this device has properties such as location, time, heat, size, etc. Therefore, defining non-physical processes and physical processes has different characteristics. PhysicalProcess is an atomic abstract for defining physical processes which have properties such as spatial reference frame for spatial and temporal reference frame for temporal coordinate reference systems (CRS). The physical location and size of objects are specified in CRS system with axes.

One object can also be relative to other outside object, thus position for learning the distance between two objects and vector property is used for specifying the relation between their positions and bounded by. The property gives information about its bounded place. These properties are reserved word in SWE.

Open Systems Interconnection (OSI) Reference Model is used for defining InterfaceDefinition which provides physical connections between processes. So InterfaceDefinition has properties that OSI has listed below in Table 1.

Table 1: OSI Layers for SensorML

<table>
<thead>
<tr>
<th>OSI layers about SensorML</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mechanical layer:</strong> Type of connector used (e.g. DB9, DB25, RJ45, USB-B)</td>
</tr>
<tr>
<td><strong>Physical layer:</strong> Provides electrical and physical characteristics of the connection including pin layouts, voltages, etc. (e.g. RS232, DSL, 802.11g)</td>
</tr>
<tr>
<td><strong>DataLink layers:</strong> Provides functional and procedural means of transferring data between network entities and correcting errors (e.g. Ethernet, 802.11, Token ring)</td>
</tr>
<tr>
<td><strong>Network layer:</strong> Provides functional and procedural means of transferring data from one source to destination via one or more networks while maintaining the Quality of Service (e.g. IP, ICMP, IPSec)</td>
</tr>
<tr>
<td><strong>Transport layer:</strong> Provides transparent transfer of data between end users and can control reliability of a given link (e.g. TCP, UDP, SPX)</td>
</tr>
</tbody>
</table>
Session layer: Controls the dialogues (sessions) between computers by establishing, managing, and terminating connections between the local and remote applications (e.g., NetBIOS, TCP session establishment).

Presentation layer: Transforms the data to provide a standard interface for the application layer (e.g., SSL, TLS, ASCII, MIDI, MPEG, SWE Common).

Application layer: Provides a means for the user to access information on the network through an application (e.g., HTTP, SMTP, FTP, XMPP, RTP).

2.4.1. COMPONENT
In real world, component can be considered as ProcessModel and it consists of a part of ProcessChain or System. It gives information about the location and interface of an object.

2.4.2. SYSTEM
System can be considered as ProcessChain. More than one physical and non-physical process are required to get desired result. In the thesis, UAVs and cameras form the considered system. UAV will be the main component and cameras will be other components. The combination of these components will form the system. The system has all necessary properties just as its own processes have. Besides its connections and interface segmentations, it also has position properties like ProcessChain.

2.5. PROCESS METADATA GROUP
Detailed information has been given by MetadataGroup. SensorML has five MetadataGroups which is shown in Figure 3.
Identifiers, classifiers, constraints, capabilities, properties, contacts, documentation sources and history properties consist of *MetadataGroup*.

All properties have their individual using places. For instance an identifier could be use when the information about names are needed whereas characteristics is used at condition such as sensors size and weight that are about just its type. General using places are listed in Table 2 below.

<table>
<thead>
<tr>
<th>General using places of Metadata Group Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Identifier:</strong> for example, shortName, longName, acronym, serialNumber, manufacturerID, partNumber.</td>
</tr>
<tr>
<td><strong>Classification:</strong> SensorType, observableType, processType, intendedApplication, or missionID</td>
</tr>
<tr>
<td><strong>Constraints:</strong> National and international securityConstraints, validTime, and legalConstraints</td>
</tr>
<tr>
<td><strong>SecurityConstraint:</strong> Based on such security definitions as the Security Banner Marking model of the Intelligence Community Information Security Marking (IC ISM) Standard.</td>
</tr>
<tr>
<td><strong>ValidTime:</strong> The validTime property indicates the time instance or time range over which this process description is valid.</td>
</tr>
<tr>
<td><strong>LegalConstraints:</strong> Privacy Act, Intellectual Property Rights, or copyrights , Boolean attributes, privacyAct, intellectualPropertyRights, and copyrights, legalConstraints</td>
</tr>
<tr>
<td><strong>Properties:</strong> The characteristics and capabilities properties take a RecordType as their value, which allows for the grouping of various properties using SWE Common DataRecord.</td>
</tr>
<tr>
<td><strong>Capabilities:</strong> For example, a particular remote sensor on a satellite might measure radiation between a certain spectral range (e.g. 700 to 900 nanometers) at a particular ground resolution (e.g. 5 meter), and with a typical spatial repeat period (e.g. 3.25 – 4.3 days).</td>
</tr>
<tr>
<td><strong>Characteristics:</strong> For example, a component may have certain physical measurements such as dimensions and weight, and be constructed of a particular material. A component may have particular power demands, or anticipated lifetime.</td>
</tr>
</tbody>
</table>
2.6. SENSORS

Every object can be defined, measured or compared with the other in a particular way. It is important to know what kind of specialty is supposed to be used so as to obtain the desired condition. These specialties can be light, pressure, motion or temperature. Some operations are to be performed in order to analyze these physical quantities. Sensors are the devices doing this conversion. If the stimulus such as light, pressure, motion or temperature is considered as an input of sensor, the observer will recognize the signal which has been received by the result. According to this type of signal, sensor can be separated as digital and analog sensors. [15] Figure 4 shows vision of digital and analog signal.

![Graphical vision of discrete and analog Signal](image)

(a) Analog Signal  
(b) Discrete Signal

*Figure 4: Graphical vision of discrete and analog Signal*

2.6.1. DIGITAL SENSORS

Digital sensors produce discrete signals. The information which is gathered via digital sensors have values that increase in particular unit. For instance, 9 bits is enough for a digital compass to symbolize the range between 0 and 359 degrees. Switches and infrared detectors are examples of these sensors which is shown in Figure 5. [15]

![Switch and Sharp IR detector](image)

(a) (b)

*Figure 5: (a) Switch  (b) Sharp IR detector*
2.6.2. ANALOG SENSORS
Analog sensors produce non-discrete signals. The output can vary from 0 volts to 5 volts. Any value can be shown in that range. The examples of these sensors are CD cells and potentiometers that are shown in Figure 6(a) and Figure 6(b) parts. [15]

![Figure 6: (a) Potentiometer Sensor (b) CdS cell wiring diagram](image)

2.7. CLASSIFICATION OF SENSORS [16]
Thanks to the technology which has constantly been developed so far, the types of sensors has also been developed. It is difficult to choose the suitable sensor which can be used for its task, thus it has become a necessity to classify sensors in accordance with its input quantity, output quantity, alimentation needs...

2.7.1. ACCELERATION, SHOCK AND VIBRATION SENSORS
There is a wide range of application areas for this kind of sensors in our daily life. When the position of an object is changed, sensor produces essential acceleration, shock or vibration, which is given as examples of different kinds of sensors such as airbag sensors or security alarms. The whole essential information about the dynamic behavior of an object is provided for the observer thanks to this result.

![Figure 7: Basic piezoelectric accelerometer construction](image)
2.7.2. BIOSENSORS

Two main components of this sensor are bioreceptor and transducer. Bioreceptor is a biomolecule. This biomolecule senses the target and transducer converts it to electric signal. Biosensor is different from other sensors because there are two components in one sensor. Therefore, the measurement of glucose in a blood sample can be done by just dipping the sensor into the blood. This reduces the time spent, thus it is useful in laboratories. Bioreceptors are generally enzymes that are capable of recognizing the target and these enzymes are used in biosensor so as to be recognized.

Figure 8 shows parts of a biosensor in a detailed way.

The connection between transducer and bioreceptor is provided by both physical and non-physical attachment. Chemical attachment can often be supplied by putting some reagents to the surface of transducer. Transducer is to not only understand the biorecognition but also convert it into a signal measured by an observer. This event often occurs when the changes which are in bioreceptor reaction are measured. For instance, the amount of enzymes also changes following the chemical changes. Therefore, it is only possible for transducer to realize the changes by measuring it.

Biosensors have eight parameters that show its characteristics. These specialties are:
1. Sensitivity

2. Selectivity

3. Range

4. Response time

5. Reproducibility

6. Detection limit

7. Life time

8. Stability

The application areas of biosensors are the measurement of metabolites such as places including emergency rooms, doctor offices which are also taken part in military and environmental monitoring.

Figure 9 shows how a biosensor works.

2.7.3. FLOW AND LEVEL SENSORS

These types of sensors are used to measure both liquid and air flow. Mass flow, volume flow, laminar flow, turbulent flow can be defined as types of flows. The most important issue which is mentioned as one of the tasks of flowing is to define the amount of substance. There are a lot of sensors which have been produced according to different technologies and these sensors have the same function which has been mentioned above. It is possible to apply some of these technologies in both air and liquid flowing situation. In fact, the
measurement of flow is the same thing as the measurement of level. These sensors reply to the question of “how much” in applications.

It is shown in Figure 10 one of flow and level sensor working principles which is provided by Pitot tube.

The method which is used for the measurement of flow can be different from the measurement of liquids, the velocity of fluid, the measurement of air and thermal changes in a particular zone. Some of the principles used in these sensors are:

2.7.3.1. THERMAL ANEMOMETERS
That it uses the principle of heat changes calculated by heat sensor from one point to another point can be a reference to the velocity of fluids. These sensors generally include two sensors because of this reason.

2.7.3.2. DIFFERENTIAL PRESSURE MEASUREMENT
It can be used for both air and liquid measurement. These types of flow meters are the most common types of flow sensors and they are mainly used for liquids. The principles of these types of sensors are measuring the pressure which has effects across each meter and taking its square root. It has two main components. The first of these components creates kinetic energy for creating differential pressure in the pipe and the second one measure this pressure and converts it to a measurable signal.
3) UNMANNED AERIAL VEHICLES SENSING MISSIONS SPECIFICATION USING SENSORML

In a SensorML program, a system constitutes the most comprehensive part of the program where all parts of an object can be thought as a system. In this thesis, five types of UAV such as helicopter UAV, quad copter UAV, airplane UAV, blimp UAV are considered different systems and inside of these systems, there are other four different systems including four different types of cameras in each UAV. These systems have been separately analyzed in this thesis.

3.1. UNMANNED AERIAL VEHICLES

3.1.1. DRAGANFLYERX6

The DraganflyerX6 which is shown in Figure 11 has six rotors which provide an autonomous flying with a hand-held controller. The data obtained from cameras are synchronized with a hand-held controller, therefore unmanned aerial vehicles are possible to be seen and administrated from a certain area where the camera has a point of view. Cameras are wireless and certain researches about these unmanned aerial vehicles have been carried out since 2006. [1]

It is compatible to work with several cameras inside this UAV with the aim of expanding application areas. These cameras are:

- 10.1 MP digital still camera with 720p video recording,
- 1080p HD video camera,
- Low light video camera,
- Thermal imaging video camera,
- Digital IP Video camera
The general application areas of DraganflyerX6 vary from security, reconnaissance, inspection, damage assessment to research, real estate promotion, or advertising. It’s a miniature helicopter that reduces time in order to be ready for the flight. Also, it has no unnecessary movable parts. When UAV is on the air, self-stabilization system automatically works and this reduces the struggle with complicated controls. And it is cheaper to gain desired results thanks to DraganflyerX6. It makes DraganflyerX6 a perfect reconnaissance machine for military applications. [1]

3.1.2. DRAGANFLYERX6 HELICOPTER TECH SPECS [2]

Figure 12 shows capabilities of DraganflyerX6 with arrows where that specialty is for. The general specifications of Helicopter listed in Table 3 below.

<table>
<thead>
<tr>
<th>Helicopter Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>o Width: 91cm (36in)</td>
</tr>
<tr>
<td>o Length: 85cm (33in)</td>
</tr>
<tr>
<td>o Top Diameter: 99cm (39in)</td>
</tr>
<tr>
<td>o Height: 25.4cm (10in)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Weight &amp; Payload</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>
- Helicopter Weight: 1,000g (35oz)
- Payload Capability: 500g (18oz)
- Maximum Gross Take-Off Weight: 1,500g (53oz)

**Flight Characteristics:**

- Unassisted visual reference required
- Max Climb Rate: 2m/s (6.5 ft/s)
- Max Descent Rate: 2m/s (6.5 ft/s)
- Max Turn Rate: 90°/second
- Approx Maximum Speed: 50km/h (30mph)
- Minimum Speed: 0km/h (0mph)
- Launch Type: VTOL (Vertical Take Off and Landing)
- Maximum Altitude ASL: 2,438m (8,000ft)
- Maximum Flight Time: Approx. 20 min (without payload)
- Approx Sound at 1m (3.28ft): 65dB
- Approx Sound at 3m (9.84ft): 60dB

### 3.1.3. DRAGANFLYERX8

![DraganflyerX8 quadcopter](image)

*Figure 13: DraganflyerX8 quadcopter*

The quad copter has eight rotors which turn it. DraganflyerX8 is shown in Figure 13. Its unique design increases the mobility in desired speed. Eight rotors carry a great deal of payload. The things which make it more sophisticated than DraganflyerX6 are its carrying capacity and safety. [3]
In order to reduce the damages in transportation and provide higher payload capability, it has been made of carbon fiber frame. If one rotor is damaged and stops working, the other rotor continues to work just as the previous situation does. [3]

It also has eleven sensors (three gyros, three accelerometers, three magnetometers, one barometric sensor, and done GPS receiver). And any person can use it after a little training. [3]

3.1.4. DRAGANFLYERX8 QUAD COPTER TECH SPECS [4]

DraganflyerX8 quad copter’s general capabilities are shown in Figure 14. General specifications are listed in Table 4 below.

<table>
<thead>
<tr>
<th>Helicopter Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Width: 87cm (36.25in)</td>
</tr>
<tr>
<td>Length: 87cm (36.25in)</td>
</tr>
<tr>
<td>Top Diameter: 106cm (41.8in)</td>
</tr>
<tr>
<td>Height: 32cm (12.6in)</td>
</tr>
</tbody>
</table>

Weight & Payload
Helicopter Weight: 1,700g (60oz)
Payload Capability: 1,000g (35oz)
Maximum Gross Take-Off Weight: 2,700g (95oz)

Flight Characteristics:
- Unassisted visual reference required
- Max Climb Rate: 2m/s (6.5ft/s)
- Max Descent Rate: 2m/s (6.5ft/s)
- Max Turn Rate: 90°/second
- Approx Maximum Speed: 50km/h (30mph)
- Minimum Speed: 0km/h (0mph)
- Launch Type: VTOL (Vertical Take Off and Landing)
- Maximum Altitude ASL: 2,438m (8,000ft)
- Maximum Flight Time: Approx. 20 min (without payload)

3.1.5. DRAGANFLY TANGO UAV AEROPLANE

This miniature aero plane in Figure 15 which called Draganfly Tango also created by Draganfly Innovations Inc in such places that can be hazardous for a real sized aero plane. The control system of this UAV is performed by just one person. It is especially used in military areas. By using tandem, wing aero plane can fly at very low speed that a normal aero plane cannot and this gives that UAV very gently standstill specialty. Also its nose design
provides the balance of UAV. Mountain takes just 3 minutes and making the duration of a flight shorter depends on the time which is spent for the preparation of a vehicle. [5]

Draganfly Tango has thermal intelligence system through the agency of four thermal sensors and some intricate circuitries to analyze the vehicle is so close to the ground or not. When the UAV is detected, it is so close to the ground that it will automatically change its position to higher place. [5]

It is fully customizable, so it is possible to add or extract parts and give wide optional using areas from close range to wide areas approximately 60 miles. [5]

### 3.1.6. DRAGANFLY TANGO AEROPLANE TECHSPECS [6]

![Figure 16: Draganfly Tango Aero plane specifications](image)

Draganfly Tango aero plane is different from the other aero plane because of its four wing design that can be seen at Figure 16 also. General specifications are listed in Table 5.

<table>
<thead>
<tr>
<th>Table 5: Draganfly Tango Aero plane specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dimensions:</strong></td>
</tr>
<tr>
<td>- Wingspan: 60 in (150 cm)</td>
</tr>
<tr>
<td>- Length: 48 in (120 cm)</td>
</tr>
<tr>
<td>- Wing Area: 900 sq. in (5625 sq. cm)</td>
</tr>
<tr>
<td><strong>Weight &amp; Payload:</strong></td>
</tr>
<tr>
<td>- Aircraft Weight: 104 oz (2.8 kg)</td>
</tr>
<tr>
<td>- Payload Capability: 40 oz (1.14 kg)</td>
</tr>
</tbody>
</table>
Maximum Gross Take-Off Weight: 144 oz (3.94 kg)

**Flight Characteristics:**

- Cruise Speed: 31 - 38 mph (50 - 60 km/h)
- Maximum Speed: 59 mph (95 km/h)
- Stall Speed: 21 mph (35 km/h)
- Launch Type: Bungee Catapult
- Maximum Altitude: 2,100 ft (640 m)
- Maximum Flight Time: 50 minutes

### 3.1.7. CIRRUS 1000 BLIMP

![Figure 17: Cirrus 1000 Blimp](image)

When the mission is so close to people, the building level of UAV and speed must also be lower. This can be handled with the help of C1000 which is a blimp type of UAV created by Skyships Limited that is shown in Figure 17. It works with electric motors or internal combustion engines at 15V can carry over 7 kg and fly from 20 minutes to 2 hours can reach airspeed of 30 knots balloon made by a polyurethane coated rip stop nylon at the aim of being most light weight form.[7]

### 3.1.8. CIRRUS 1000 BLIMPECH SPECS [8]

Specification of Cirrus 1000 Blimp listed below in Table 6. This product is created another company from all others.
Table 6: Cirrus 1000 Blimp Tech Specs

Dimensions:
- Length: 9.50 m
- Maximum diameter: 2.50 m
- Height (including gondola): 2.80 m

Weight & Payload:
- Gross weight: 30 kg
- Payload Capability: 7 kg msl

Flight Characteristics:
- Operating ceiling: 120 m (400ft) - see below
- Envelope fabric: Polyurethane coated rip stop nylon
- No. Gores: 8
- Volume: 30 cu m / 1000 cu feet
- Surface Area: 58 sq m
- Lifting Gas: 95% Helium (Balloon Gas)
- Valves: Manual and overpressure
- Engines: 2x 1000W Brushless electric motors

3.2. CAMERAS

3.2.1. IP VIDEO CAMERA

Figure 18: IP Video Camera

This camera is a wireless camera like the others and the desired information (photograph or video) is sent on a Wi-Fi link which is shown in Figure 18. Connection from any laptop to
wireless base station will provide to see what helicopters camera can see at that time. It’s not in NTSC or pal mode. In other words, resolution is not limited like in analog video format. [9]

3.2.2. DIGITAL STILL CAMERA WITH 720P VIDEO RECORDING

Figure 19: Digital Still Camera with 720p Video Recording

Figure 19 shows photo of Digital Still Camera with 720p Video Recording. As it name when a detailed image with higher quality is needed, this camera will be the perfect choice for it. It has specialties such as 10.1 Megapixel resolution, filming 720p motion video, 5.8 GHz wireless connection and onboard SD/SDHC memory card. [10]

3.2.3. LOW LIGHT CAMERA

Figure 20: Low light Camera

Low light camera (which is shown in Figure 20) works nearly as 0.0001 lux ambient light that means it will be possible to gain clear photos as black and white form almost in dark environment. Data stored on-board DVR and send approximately 5.8 GHz wireless connection. [11]
3.2.4. THERMAL INFRARED CAMERA

![Thermal infrared Camera](image)

Figure 21: Thermal infrared Camera

Figure 21 shows photo of thermal camera. When the task is about searching people in an area and detecting some events like fire in a forest, this camera will be the first choice. It is a powerful thermal camera that can detect a human from its body heat. It transmits its data over 5.8 GHz wireless connections. [12]

3.2.5. MICRO ANALOG VIDEO CAMERA

![Micro Analog Video Camera](image)

Figure 22: Micro Analog Video Camera

This camera is the smallest one from the other cameras that is shown in Figure 22. It’s in analog form and has specialties like 480 interlaced lines of full color analog motion video at a frame rate of 30fps. It transmits its data approximately over a 5.8 GHz wireless connection like other cameras. [13]
2.6. VIDEO & STILL CAMERA SYSTEM SPECIFICATIONS [14]

Table 7: Video & Still Camera System Specifications

<table>
<thead>
<tr>
<th>Camera Brand</th>
<th>Features</th>
<th>Benefits</th>
<th>CCD / MOS / Sensor</th>
<th>Weight of Camera Alone:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Panasonic Luminix DMC-LX3</td>
<td>- 10.1 Megapixel</td>
<td>- Professional quality digital still photographs</td>
<td>1/1.63 inch CCD</td>
<td>8.0oz / 227g</td>
</tr>
<tr>
<td></td>
<td>- 24mm Wide-Angle Lens</td>
<td></td>
<td></td>
<td>Camera &amp; Mount:</td>
</tr>
<tr>
<td></td>
<td>- 2.5x Optical Zoom</td>
<td></td>
<td>11.6oz / 333g</td>
<td>10.6oz / 300g</td>
</tr>
<tr>
<td></td>
<td>- 964x2736 Max Image Size</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Image Stabilization</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Saves to SD/SDHC/CompactMemory Card</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Can Film 720p Video</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Panasonic HDC-SD9</td>
<td>- Full HD 1020x1080 Progressive</td>
<td>- Professional quality digital still photographs</td>
<td>1/6 inch 3CCD</td>
<td>11.6oz / 333g</td>
</tr>
<tr>
<td></td>
<td>- 25mm Equivalent Leica Lens</td>
<td></td>
<td></td>
<td>Camera &amp; Mount:</td>
</tr>
<tr>
<td></td>
<td>- 10x Optical Zoom</td>
<td></td>
<td>14.2oz / 406g</td>
<td>14.2oz / 406g</td>
</tr>
<tr>
<td></td>
<td>- 1/4 inch 3CCD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Image Stabilization</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Records to SD/SDHC/CompactMemory Card</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Watac WAT-9024-2 Ultimate</td>
<td>- 768x512 Effective Pixels</td>
<td>- Low light motion video</td>
<td>1/2 inch CCD</td>
<td>5.8oz / 153g</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Clear video where other cameras would only show black</td>
<td></td>
<td>Camera &amp; Mount:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8.4oz / 231g</td>
<td>8.4oz / 231g</td>
</tr>
<tr>
<td>FUR Photon 320</td>
<td>- Photon 320</td>
<td>- Thermal infrared motion video</td>
<td>320 Camera Alone:</td>
<td>5.4oz / 153g</td>
</tr>
<tr>
<td></td>
<td>- 3246x966 Effective Pixels</td>
<td></td>
<td></td>
<td>Camera &amp; Mount:</td>
</tr>
<tr>
<td></td>
<td>- 1/2 inch CCD</td>
<td></td>
<td>8.0oz / 228g</td>
<td>8.0oz / 228g</td>
</tr>
<tr>
<td></td>
<td>- 570 TV Lines Interlaced</td>
<td></td>
<td></td>
<td>Camera &amp; Mount:</td>
</tr>
<tr>
<td></td>
<td>- 18.25mm Lens</td>
<td></td>
<td>8.0oz / 228g</td>
<td>8.0oz / 228g</td>
</tr>
<tr>
<td></td>
<td>- Multifunctional: 10.1MP Pictures and</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>11.6oz / 333g</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>14.2oz / 406g</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Micro Board Camera</td>
<td>- 768x512 Effective Pixels</td>
<td>- Analog motion video</td>
<td>1/3 inch CCD</td>
<td>0.42oz / 12g</td>
</tr>
<tr>
<td></td>
<td>- 480 TV Lines Interlaced</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Small and Light</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Uncooled LWIR Thermal Imager</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Sensor: Vox</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Microboimeter</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Forward Looking Infrared</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- 7.2 to 13.5μm Spectral Band</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Multifunctional: 2CCD cameras</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Can Film 720p HD Video</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: The DMC-LX3 is modified for remote zoom, shutter, and battery.
Table 7 represents general specifications of five camera sensors. Same specifications separate cameras with different values that help to clarify their individual using places. Take a closer look to those specialties:

Includes: All types of cameras includes anti-vibration camera mount to fit determined UAV properly. Also it is important to take images from UAV by the help of antenna and video transmitter.

Benefits: It’s the most important part to recognize cameras identical properties. It is seen from Table 7 that Low light camera is capable to see dark places. It makes Low light camera

<table>
<thead>
<tr>
<th>Color</th>
<th>Full Color</th>
<th>Full Color</th>
<th>Monochrome</th>
<th>Full Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>Onboard Digital Video Recording Mode</td>
<td>Format QuickTime Motion Photo JPEG 1280x720, 24fps, 24Mbps</td>
<td>Format AVCHD 1080i 17Mbps w Variable Bitrate 1080p 13Mbps w Variable Bitrate 1920 x 1080p 6Mbps w Variable Bitrate</td>
<td>Monochrome False Color Option</td>
<td>Polarity Control</td>
</tr>
<tr>
<td>TV Out Playback Mode</td>
<td>NTSC</td>
<td>PAL</td>
<td>Selectable</td>
<td>NTSC</td>
</tr>
<tr>
<td>Onboard Video / Image Storage</td>
<td>SD / SDHC / MMC Memory Card</td>
<td>JPEG / Motion JPEG</td>
<td>SD / SDHC Memory Card</td>
<td>AVCHD Format</td>
</tr>
<tr>
<td>Zoom</td>
<td>2.5x Optical Zoom</td>
<td>8x Digital Zoom</td>
<td>Remote Zoom Using Handheld Controller</td>
<td>10x Optical Zoom</td>
</tr>
<tr>
<td>Lens</td>
<td>Leica DC Vario-Summicron, 24 - 60mm Wide-Angle Lens</td>
<td>35mm Equivalent Leica Lens</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Focal Length</td>
<td>5.1 to 12.8mm</td>
<td>3.0 to 30mm</td>
<td>4 to 12mm</td>
<td></td>
</tr>
<tr>
<td>Onboard Image Stabilization</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>ISO / Minimum Illumination</td>
<td>80 to 6400 ISO</td>
<td>5 Lux</td>
<td>0.0001 Lux</td>
<td></td>
</tr>
</tbody>
</table>
suitable when the mission time is night. Thermal camera is good option search and rescue missions. It can be used for instance a human is searched in a forest except the other cameras. Micro analog camera is the smallest camera so if camera wanted to be hide from enemies Micro analog camera will be the perfect choice for it.

Weight: It is clear from the Table 7 that Micro Analog camera has least weight. Because of that Micro analog camera can be used when the payload capability take importance.

Color: Low light camera is monochrome except the other cameras. Also thermal camera has false color option that gives another color instead of its real color because of thermal cameras ability to seeing images in dark not from its light from its heat. Thus if colorful images are wanted it is not a good option to choose thermal or low light camera.

Zoom: As it can be considered HD camera has biggest zoom capacity. So when the target is far away and detailed image is wanted HD camera will be the first option for suitable UAV.

Lens: Lens is related with zooming. HD camera lenses have the capacity of seeing most detailed image.

ISO/ minimum illumination: Minimum light for a clear photo so Low light camera decreases its values as lowest as it could that is shown in the table.

3.3. IMPLEMENTATION
This thesis separated into two parts. In the first part of the thesis, systems representing UAVs (quad copter, helicopter, aero plane and blimp) are arranged by using a SensorML program. These systems are shown in Figure 23 and Figure 24. Each UAV has possibility to have five camera sensors (low light, thermal, micro analog, HD, normal) inside the vehicles. These sensors have been designed to be used in various missions and the main reason why different unmanned aerial vehicles are used is to find the most suitable UAV for the environmental conditions of mission which has been chosen.
For instance, in a mission that happens at night and if it takes a lot of time and it must be fast, the UAV will probably be an aero plane with these four UAV and the camera must be thermal or low light camera from five camera sensors.

Datasheets including UAV and the general characteristics of camera sensors are used to determine this situation. The Information Inside datasheets transform into databases in XML form by using SensorML. Basically, every UAV has five camera sensors and this makes these databases hierarchic databases stored in XML form by using SensorML.

In the second part of the thesis, an interrogation system has been designed with the aim of using the databases with the description of the sensors and the UAVs. The main concept of
creating interrogation system is to calculate the best option in specific mission for UAV and cameras.

Each question has several choices which listed as checkboxes or radio buttons. Previously, the chosen mission’s environmental preferences are determined. For instance, if mission happens in the morning and detailed photograph is needed in long distance, it has to be chosen daylight in question about time, the highest detailed camera in question about resolution, and for long distance, the vehicle whose capacity is the highest position in the air must be chosen. There are approximately 15 questions about UAVs and 15 questions about camera sensors and each of the questions include specialties about vehicles. After all questions have been answered, PHP code transforms data which are sent by HTML page, runs the code with the aim of finding the most suitable UAV and camera for desired mission and returning the answer in second HTML page.

Databases done by SensorML program is used for the operations between PHP and HTML codes. Data taken from HTML page is sent to PHP code and data search the corresponding data in databases by using this code. When data match between databases and data sent from HTML document, it requires UAV or camera sensor to give itself a point.

After all this operation, it will be easier to choose which UAV and camera sensor fits into the situation of mission.

The first part of the thesis has been explained below in a detailed way.

3.3.1. DATABASE
The main concept is the creation of XML databases which are UAV and camera sensor data that are given from their datasheets by SensorML program.

Each UAV (helicopter, aero plane, quad copter and blimp) carries thermal, micro analog, low light and HD cameras. Five camera sensors will be five separated systems and it will be combined with UAV at final and build up the main system.

General vision of SensorML program has been shown below. Every vehicle also consists of a system and this system has specialties such as keywords, identification, classification, valid time, security constraints, legal constraint, characteristics, capabilities, documentation,
Low Light camera is chosen to show parts of SensorML Program.

### 3.3.1.1. KEYWORDS

This part contains main terms about the vehicles such as its name, type, differences from other vehicles.

Keywords generally show the user how to work on it. Listing 3.1 shows Low light cameras keyword part which is written in SensorML.

```
1) <sml:keywords>
2) <sml:KeywordList>
3) <sml:keyword>Low Light</sml:keyword>
4) <sml:keyword>Camera</sml:keyword>
5) <sml:keyword>Watec</sml:keyword>
6) <sml:keyword>WAT-902H2</sml:keyword>
7) </sml:KeywordList>
8) </sml:keywords>
```

*Listing 3.1: Keywords (SensorML Code)*

In lines between 3 and 6 basic words clarify low light camera being a low light camera and its name “Watec” gives general hints about what the user working on it.

### 3.3.1.2. IDENTIFICATION

Identification part symbolizes analyze sensor. As its name identification include what identifies this vehicle such as its name (long or short name), manufacturer name, unique ID, mission ID, instrument name. Listing 3.2 clarifies that with the example of Low light camera’s identical specifications such as its name and its manufacturer name below.
In lines 30, 35, 40 low light cameras identical values such as short name, long name, manufacturer name is specified and in lines 32, 37, 42 determined values for that specialties are given.

3.3.1.3. CLASSIFICATION
Every sensor has its own usage areas and these areas are wide-ranging, therefore there is a need to classify them. Classification part plays an important role to classify sensor systems. It generally shows the specialties such as sensor type, platform type, system type, application areas. Listing3.3. shows determined sensor is a camera and it is a platform type sensor.
Sensor type and platform type are chosen as classifiers which are shown in lines 67 and 72 and their values are given in lines 69 and 74 as platform for platform type and camera for sensor type.

3.3.1.4. CHARACTERISTICS
Characteristics part shows the measurable specialties of vehicles in 3D space. If it is a camera, characteristics are resolution, minimum illumination, TV-lines interlaced, color or if it is a UAV, its width, length, front diameter, height, payload capability, maximum gross take-off weight symbolizes this area. General measurable specifications of DraganflyX8 quadcopter such as its size, weight payload capabilities etc. are shown in Listing 3.4. below.
<swe:DataRecord>
  <swe:field name="Length">
    <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Length">
      <swe:uom code="cm" xlink:href=""/>
      <swe:value>87</swe:value>
    </swe:Quantity>
  </swe:field>
  <swe:field name="Top Diameter">
    <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:TopDiameter">
      <swe:uom code="cm" xlink:href=""/>
      <swe:value>106</swe:value>
    </swe:Quantity>
  </swe:field>
  <swe:field name="Height">
    <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Height">
      <swe:uom code="cm" xlink:href=""/>
      <swe:value>32</swe:value>
    </swe:Quantity>
  </swe:field>
</swe:DataRecord>

<sml:characteristics name="Weight and Payload">
  <swe:DataRecord>
    <swe:field name="Weight">
      <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:HelicopterWeight">
        <swe:uom code="g" xlink:href=""/>
        <swe:value>1700</swe:value>
      </swe:Quantity>
    </swe:field>
    <swe:field name="Payload Capability">
      <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:PayloadCapability">
        <swe:uom code="g" xlink:href=""/>
        <swe:value>1000</swe:value>
      </swe:Quantity>
    </swe:field>
  </swe:DataRecord>
</sml:characteristics>
DraganflyX8 separates two main characteristic parts that are helicopter size dimensions and weight and payload which is shown in line 92 and 120 written SensorML with reserved words “sml:characteristics name”. Helicopter size dimensions specialties are width, length, top diameter, high is defined in lines 94, 106 and 112 in SensorML “swe:field name”. Besides weight and payload specialties are weight, payload capability also in line 122, 128, 134. All their values are represented with reserved words “swe:value” and these values type “swe:uom code”. As an example of that reserved words are shown in lines 136 and 137 Maximum gross take-off weight is 2700 gram.

3.3.1.5. CAPABILITIES
The usefulness of vehicle has been shown in capabilities area. This place is generally formed by the capability of vehicle and its limitations. Specialties of this area are some kind of preferences like. Chosen UAV is DraganflyX8 to represent capabilities specialty in Listing 3.5. below.
<swe:field name="Max Descent Rate">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxDescentRate">
    <swe:uom code="m/s" xlink:href=""/>
    <swe:value>2</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="Max Turn Rate">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxTurnRate">
    <swe:uom code="degree/second" xlink:href=""/>
    <swe:value>90</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="Approx Maximum Speed">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ApproxMaximumSpeed">
    <swe:uom code="km/h" xlink:href=""/>
    <swe:value>50</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="Minimum Speed">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MinimumSpeed">
    <swe:uom code="" xlink:href=""/>
    <swe:value>0</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="Maximum Altitude ASL">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumAltitudeASL">
    <swe:uom code="m" xlink:href=""/>
    <swe:value>2438</swe:value>
  </swe:Quantity>
</swe:field>
Max Climb Rate in line 152 with represented code with "swe:field name", value of that specification in line 155 as 2 with represented code with "swe:value" and type of value as m/s in line 154 with represented code with "swe:uom code" in SensorML.

As it is seen reserved words are same except one which is shown in line 150 as Flight specifications of that capabilities name is written SensorML with reserved word "sml:capabilities name". Logic is same for other capability specifications that can be seen in the Listings 3.5.

3.3.1.6. CONNECTION LIST
The connection list area is about who produced the vehicle and how to be connected with the producers. The specialties of this part are individual name, organization name, contact
info, phone, address, administrative area, postal code, country. All contact lists will be same as seen in Listing 3.5. for Draganfly products.

```
252)  <sml:contact>
253)   <sml:ResponsibleParty>
254)   <sml:individualName>Draganfly</sml:individualName>
255)   <sml:organizationName>Draganfly Innovations Inc.</sml:organizationName>
256)   <sml:contactInfo>
257)    <sml:phone>
258)     <sml:facsimile>306-955-9907</sml:facsimile>
259)    </sml:phone>
260)    <sml:address>
261)     <sml:city/>
262)     <sml:administrativeArea/>
263)    </sml:postalCode/>
264)    <sml:country/>
265)    <sml:electronicMailAddress>info@draganfly.com</sml:electronicMailAddress>
266)   </sml:address>
267)   </sml:contactInfo>
268)   </sml:ResponsibleParty>
269)  </sml:contact>
```

Listing 3.6: Contact List (SensorML Code)

All connection List parts are same for products DraganflyX8, Draganflyer Tango, DraganflyX6 and All camera types as a result being a Draganfly innovations product. It can be seen in the listing 3.6 some reserved word “sml:individualName” as product company individual name in Line 254, “sml:organizationName” as organization name in line 255, “sml:facsimile” as facsimile in line 258, “sml:electronicMailAddress” as e-mail address in line 265.

### 3.4. SENSING MISSION MANAGEMENT PROGRAM

In this part of the thesis, a program that uses the database with the specification of UAVs and sensors is created, which is responsible for the selection of the best matching UAV
platform and sensor device, which has the goal to be used as a sensing mission management program. The program interface is in a format of questionnaire form and consists of 15 questions about UAVs and 15 questions about camera sensors that are represented in Figure 26 and Figure 27.

Questions part is in HTML form and designed as a web page. A question has several options in radio buttons form. After all appropriate answer chosen HTML page mission will be finished when submit button pressed.

PHP code takes all determined data which send from HTML page. PHP code operates this data and databases which is created with using SensorML program takes role at this part. Totally this thesis has 5 UAV and 5 camera sensors databases for searching appropriate UAV and camera sensors in an given mission environmental conditions. PHP code algorithm works as shown below:
1. Required data taken from HTML form. Answers which taken from Listboxes are assigned to the array variable called $questions.(see Line 21)

2. After submit button is pressed all questions separately checked if it has a POST[] value or not . If desired condition is not succeeded an error message will be shown that is shown lines between. (see Lines between 37 - 44)

3. Algorithm takes role after all questions checked. In fact each of these questions is related a specific area. That area is in capabilities or characteristics part because of questions specifications.

4. Characteristics and capabilities has Quantity attribute that will help to find the most suitable UAV. Quantity area has definition, value, uom code attributes. (marked yellow in Listing 3.7.) Quantity’s definitions have identical names such as 'urn:ogc:def:property:OGC:1.0:Width'. Definitions are related with what the question is about so each of these questions definition values registered in an array called $search_questions in Line 2. For instance first question is “Width” and its related definition is 'urn:ogc:def:property:OGC:1.0:Width' in Line 3. 15 questions related definition part is registered array value search_questions separately. (see lines between 47-58)

5. All Quantity parts will be scanned and if its value matches with POST['questionvalue'] and its definition part matches $search_text it is clear that the chosen answer is in that database and related with that UAV. Each UAV has a counter and that increases when conditions are succeed.SimpleXMLElement function seperates Xml file into its parts.

6. After all databases searched biggest value in counter ascertained with max function and most suitable UAV for that condition is show with echo function. (see lines between 112 - 120) All of these lines are shown in Listing 3.8
```php
<?php
$search_questions = array('urn:ogc:def:property:OGC:1.0:Width',
....
$questions = array('question1' => 'list1', 'question2' => 'list2',
....
$counter_draganflyx6=0;
....
if(isset($_POST['submit'])) {
  foreach($questions as $questionkey => $questionvalue) {
    $questionpost = isset($_POST[$questionvalue]) ? $_POST[$questionvalue] : '';
    if($questionkey == empty($questionpost)) {
      $errormsg = 'Please select an answer for '. $questionkey;
      echo $errormsg . '<br />';
    } } }
//if an question is not checked this error message will be shown

foreach ($questions as $questionkey => $questionvalue) {
  foreach ($search_questions as $search_text) {
    $sml:SensorML= new SimpleXMLElement('draganflyx6.xml', null, true); //all xml files wanted to be separate its elements
    $quantity_characteristics=$sml:SensorML>xpath('sml:member/sml:System/sml:characteristics/swe:DataRecord/swe:field/swe:Quantity');
    foreach ( $quantity_characteristics as $swe:Quantity ) {
      if ($swe:Quantity['definition']== $search_text && $swe:Quantity->swe:value== $_POST[$questionvalue])
        $counter_draganflyx6++; }   // if search text and question text is available increase counter
    $quantity_capabilities=$sml:SensorML>xpath('sml:member/sml:System/sml:capabilities/swe:DataRecord/swe:field/swe:Quantity');
    foreach ( $quantity_capabilities as $swe:Quantity ) {
      if ($swe:Quantity['definition']== $search_text && $swe:Quantity->swe:value== $_POST[$questionvalue])
        $counter_draganflyx6++;  }    // if search text and question text is available increase counter
  } 
}]
```

40
Sometimes one attribute can be matched more than one database. At situations like that determined databases chosen together.

Listing 3.8.: UAV Questionaire Code (Just for first question as given example)

```
foreach($questions as $questionkey => $questionvalue) {
    $HD_Video= file_get_contents("HDVideo.xml");
    if( !empty( preg_match($_POST[$questionvalue], NULL , $HD_Video)))
        $counter_HD_Video++;
}}
```

Listing 3.9.: Camera Questionaire Code (differences from UAV questionnaire)
Figure 27: Web page about Camera Questions

Figure 27 shows camera questionnaire prepared same algorithm with UAV questionnaire. It just uses another function to reach result (pregmatch). Also this time pregmatch function gives enough ability to find POST['questionvalue'] and $search_text separately (see Listing 3.9.). $search_texts this time looks generally field areas which is same with question name (not required definitions of urns just question name is enough).

Thermal Infrared camera is the best option for that condition

DrangaflyerX6 is chosen as most suitable UAV for that condition

Figure 28: Result pages
4) RESULTS

The main results of the thesis can be divided in two groups: Supporting Database and Mission Management Software.

Supporting Database

The first part is composed of the data acquisition about UAVs and sensors to be employed in sensing missions to build a database to support the final intended application. The following list represents the main achievement in this part:

1. Necessary information is obtained about sensor, sensor types, UAV and UAV types.
2. Necessary information is obtained about Sensor ML and SensorML program rules are found out.
3. Necessary UAVs and camera are determined for this thesis.
4. Specifications of that UAVs and cameras which is written in the datasheets that belong to these vehicles gathered in corresponding manner.
5. That gathered information is written in SensorML program to create necessary databases in XML format.

The second part is represented by the development of software to be used in sensing mission management, by selecting the best fit between a mission and a UAV plus sensor to handle it. The following list provides the main achievements in this part.

1. A web page is created which consist of totally 30 questions about chosen mission specialties by using HTML.
2. Databases that created by using SensorML is used in codes that is written in PHP.
3. And necessary program is written.

Figure 26 to 28 show screenshots of the web pages created for the mission management software, with which the user configures the missions.
Automation is one of the most important research and industrial topics nowadays, in which the goal is that systems can perform jobs automatically without any human directives. The use of autonomous vehicles is an important part of this trend, in which autonomous automatic vehicles can perform jobs in which the employment of humans could be dangerous or too expensive, for example in sensing missions like those that aim to gather information about enemies in the battle field. In this context, there are several types of product such as unmanned aerial vehicles and also its types changing and developing recently. An unmanned aerial vehicle can be a helicopter, a quad copter, an aero plane or a blimp in being condition that differ in its using areas, used time or speed. With the aim of gaining desired information there is a need for different type of sensors in these UAVs. Thus, the selection of a UAV and the sensors to be used is a big issue, which is handled in this thesis.

After the most appropriate UAV is determined in specific mission it is time to choose which kind of sensor will be accompany to the UAV. There can be confusion when the number of sensor types and UAV is increasing.

This thesis is written with the goal of provide a contribution to sensing missions management by providing a software support by means of saving the datasheet information about UAV and cameras in databases by using SensorML program and then using that data to choose most suitable UAV and camera in desired situation. Increasing technology will also change types of UAVs and sensors which will be provide using those autonomic systems in larger areas. But for taking it under control there is a need for classification. Clearly the content of this thesis contributes in the direction of organizing and make this tasks more user friendly.

The thesis focused on the use of SensorML to develop the software, as SensorML is a language specifically used for sensing systems, with concepts that easier the development. An additional goal of the thesis is exactly to show this feature, the usability of SensorML.
REFERENCES

1. DRAGANFLYERX6. Copyright © 2009 Draganfly Innovations Inc. Available in:

2. DRAGANFLYERX6 HELICOPTER TECH SPECS. © 2009 Draganfly Innovations Inc. Available in:

3. DRAGANFLYERX8. © 2009 Draganfly Innovations Inc. Available in:

4. DRAGANFLYERX8 HELICOPTER TECH SPECS. © 2009 Draganfly Innovations Inc.

5. DRAGANFLY TANGO UAV AEROPLANE. © 2009 Draganfly Innovations Inc. Available in:

6. DRAGANFLY TANGO AEROPLANE TECH SPECS. © 2009 Draganfly Innovations Inc. Available in:

7. CIRRUS 1000 BLIMP. Cirrus Innovations Inc. Available in:

8. CIRRUS 1000 BLIMP TECH SPECS. Cirrus Innovations Inc. Available in:

9. IP VIDEO CAMERA. © 2009 Draganfly Innovations Inc. Available in:

10. DIGITAL STILL CAMERA WITH 720P VIDEO RECORDING. © 2009 Draganfly Innovations Inc.
    Available in:

11. LOW LIGHT CAMERA. © 2009 Draganfly Innovations Inc. Available in:
12. THERMAL INFRARED CAMERA. © 2009 Draganfly Innovations Inc. Available in:

13. MICRO ANALOG VIDEO CAMERA. © 2009 Draganfly Innovations Inc. Available in:

14. VIDEO & STILL CAMERA SYSTEM SPECIFICATIONS. Draganflyer X6 Helicopter Camera Options-Copyright © 2009 Draganfly Innovations Inc. 1-800-979-9794 / 306-955-9907 info@draganfly.com. Available in:

15. SENSORS. Available in:

16. CLASSIFICATION OF SENSORS.
Sensor technology handbook, Jon S. Wilson p. 1,137,161,237

17. SENSOR MODEL LANGUAGE (SENSORML). Available in:
OpenGIS® Sensor Model Language (SensorML) Implementation Specification
APPENDIX

UAV XML DATABASES
1. DRAGANFLYER X6

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<sml:SensorML rng:version="1.0.1"
xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ism="urn:us:gov:ic:ism:v2"
xmlns:rng="http://relaxng.org/ns/structure/1.0"
xmlns:sml="http://www.opengis.net/sensorML/1.0.1"
xmlns:swe="http://www.opengis.net/swe/1.0.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xng="http://xng.org/1.0">
    <sml:member>
        <sml:System gml:id="x6">
            <gml:description>Draganflyer X6 Helicopter</gml:description>
            <sml:keywords>
                <sml:KeywordList>
                    <sml:keyword>UAV</sml:keyword>
                    <sml:keyword>HELICOPTER</sml:keyword>
                    <sml:keyword>DRAGANFLYERX6</sml:keyword>
                </sml:KeywordList>
            </sml:keywords>
            <sml:identification>
                <sml:IdentifierList>
                    <sml:identifier name="Short Name">
                        <sml:Term definition="urn:ogc:def:identifier:OGC:shortName">
                            <sml:value>Draganflyer X6</sml:value>
                        </sml:Term>
                    </sml:identifier>
                    <sml:identifier name="Designer Name">
                        <sml:Term definition="urn:ogc:def:identifier:OGC:designerName"/>
                    </sml:identifier>
                </sml:IdentifierList>
            </sml:identification>
        </sml:System>
    </sml:member>
</sml:SensorML>
```
<sml:value>Draganfly Innovations Inc. Innovative UAV Aircraft & Aerial Video Systems</sml:value>
</sml:Term>
</sml:identifier>
sml:identifier name="Builder Name">
sml:Term definition="urn:ogc:def:identifier:OGC:builderName">
sml:value>Draganfly Innovations Inc.</sml:value>
</sml:Term>
</sml:identifier>
sml:identifier name="UniqueID">
sml:Term definition="urn:ogc:def:identifier:OGC:uniqueID">
sml:value>x6</sml:value>
</sml:Term>
</sml:identifier>
</sml:IdentifierList>
</sml:identification>
sml:classification>
sml:ClassifierList>
sml:classifier name="Sensor Type">
sml:Term definition="urn:ogc:classifier:sensorType">
sml:value>Platform</sml:value>
</sml:Term>
</sml:classifier>
sml:classifier name="Platform Type">
sml:Term definition="urn:ogc:classifier:platformType">
sml:value>Uav</sml:value>
</sml:Term>
</sml:classifier>
</sml:ClassifierList>
</sml:classification>
sml:securityConstraint>
sml:Security ism:classification="U" />
</sml:securityConstraint>
sml:characteristics name="Helicopter Size Dimensions">
swe:DataRecord>
swe:field name="Width">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Width">
swe:uom code="cm" xlink:href="" />
swe:value>91</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Length">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Length">
swe:uom code="cm" xlink:href="" />
swe:value>85</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Top Diameter">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:TopDiameter">
swe:uom code="cm" xlink:href="" />
swe:value>99</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Height">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Height">
swe:uom code="cm" xlink:href="" />
swe:value>25.4</swe:value>
</swe:Quantity>
</swe:field>
</swe:DataRecord>
<sml:characteristics name="Weight and Payload">
  swe:DataRecord>
  swe:field name="Weight">
    swe:Quantity definition="urn:ogc:def:property:OGC:1.0:HelicopterWeight">
      swe:uom code="g" xlink:href="" />
    swe:value>1000</swe:value>
  </swe:field>
  swe:field name="Payload Capability">
    swe:Quantity definition="urn:ogc:def:property:OGC:1.0:PayloadCapability">
      swe:uom code="g" xlink:href="" />
    swe:value>500</swe:value>
  </swe:field>
  swe:field name="Maximum Gross Take-Off Weight">
    <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumGrossTake-OffWeight">
      swe:uom code="g" xlink:href="" />
    swe:value>1500</swe:value>
  </swe:field>
  </swe:DataRecord>
</sml:characteristics>

<sml:capabilities name="Flight">
  swe:DataRecord>
  swe:field name="Max Climb Rate">
    swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxClimbRate">
      swe:uom code="m/s" xlink:href="" />
    swe:value>2</swe:value>
  </swe:field>
  </swe:DataRecord>
</sml:capabilities>
<swe:field name="Max Descent Rate">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxDescentRate">
    <swe:uom code="m/s" xlink:href="" />
    <swe:value>2</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="Max Turn Rate">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxTurnRate">
    <swe:uom code="degree/second" xlink:href="" />
    <swe:value>90</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="Approx Maximum Speed">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ApproxMaximumSpeed">
    <swe:uom code="km/h" xlink:href="" />
    <swe:value>50</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="Minimum Speed">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MinimumSpeed">
    <swe:uom code="" xlink:href="" />
    <swe:value>0</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="Maximum Altitude ASL">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumAltitudeASL">
    <swe:uom code="m" xlink:href="" />
    <swe:value>2438</swe:value>
  </swe:Quantity>
</swe:field>
<swe:field name="Approx Sound at 1m">
<swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ApproxSoundat1m">
<swe:uom code="dB" xlink:href="" />
<swe:value>65</swe:value>
</swe:Quantity>
</swe:field>

<swe:field name="Approx Sound at 3m">
<swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ApproxSoundat3m">
<swe:uom code="dB" xlink:href="" />
<swe:value>60</swe:value>
</swe:Quantity>
</swe:field>
</swe:DataRecord>
</sml:capabilities>
sml:contact>
sml:ResponsibleParty>
sml:individualName>Draganfly Innovations Inc.</sml:individualName>
sml:contactInfo>
sml:phone>
sml:voice>1-800-979-9794</sml:voice>
sml:facsimile>1-306-955-9906</sml:facsimile>
</sml:phone>
<sml:address>
  <sml:deliveryPoint>Draganfly Innovations Inc. 2108 St. George Avenue Saskatoon, SK S7M0K7 Canada</sml:deliveryPoint>
</sml:address>

<sml:contactInfo>
  <sml:ResponsibleParty>
    <sml:contact>
      <sml:System>
        <sml:member>
          <sml:SensorML>
            <sml:member>
              <sml:System gml:id="x6">
                <gml:description>Draganflyer X8 Quadcopter</gml:description>
                <sml:keywords>
                  <sml:KeywordList>
                    <sml:keyword>UAV</sml:keyword>
                    <sml:keyword>QUADCOPTER</sml:keyword>
                    <sml:keyword>DRAGANFLYERX8</sml:keyword>
                  </sml:KeywordList>
                  <sml:identification>
                    <sml:IdentifierList>2.DRAGANFLYERX8</sml:IdentifierList>
                  </sml:identification>
                </sml:keywords>
              </sml:System>
            </sml:member>
          </sml:SensorML>
        </sml:member>
      </sml:System>
    </sml:contact>
  </sml:ResponsibleParty>
</sml:contactInfo>

sml:identifier name="Short Name">
  sml:Term definition="urn:ogc:def:identifier:OGC:shortName">
  sml:value>Draganflyer X8</sml:value>
</sml:Term>
</sml:identifier>
sml:identifier name="Designer Name">
  sml:Term definition="urn:ogc:def:identifier:OGC:designerName">
  <sml:value>Draganfly Innovations Inc. Innovative UAV Aircraft & Aerial Video Systems</sml:value>
</sml:Term>
</sml:identifier>
sml:identifier name="Builder Name">
  sml:Term definition="urn:ogc:def:identifier:OGC:builderName">
  <sml:value>Draganfly Innovations Inc.</sml:value>
</sml:Term>
</sml:identifier>
sml:identifier name="UniqueID">
  sml:Term definition="urn:ogc:def:identifier:OGC:uniqueID">
  <sml:value>x8</sml:value>
</sml:Term>
</sml:identifier>
</sml:IdentifierList>
</sml:identification>
sml:classification>
sml:ClassifierList>
sml:classifier name="Sensor Type">
  sml:Term definition="urn:ogc:classifier:sensorType">
  <sml:value>Platform</sml:value>
</sml:Term>
</sml:classifier>
sml:classifier name="Platform Type">
  sml:Term definition="urn:ogc:classifier:platformType">
    sml:value>Uav</sml:value>
  </sml:Term>
  </sml:classifier>
</sml:ClassifierList>
</sml:classification>
sml:securityConstraint>
  sml:Security ism:classification="U" />
</sml:securityConstraint>
sml:characteristics name="Helicopter Size Dimensions">
swe:DataRecord>
  swe:field name="Width">
    swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Width">
      swe:uom code="cm" xlink:href="" />
    swe:value>87</swe:value>
  </swe:Quantity>
  </swe:field>
  swe:field name="Length">
    swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Length">
      swe:uom code="cm" xlink:href="" />
    swe:value>87</swe:value>
  </swe:Quantity>
  </swe:field>
  swe:field name="Top Diameter">
    swe:Quantity definition="urn:ogc:def:property:OGC:1.0:TopDiameter">
      swe:uom code="cm" xlink:href="" />
    swe:value>106</swe:value>
  </swe:Quantity>
  </swe:field>
<swe:field name="Height">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Height">
    <swe:uom code="cm" xlink:href=""/>
    <swe:value>32</swe:value>
  </swe:Quantity>
</swe:field>
</swe:DataRecord>
</sml:characteristics>
<sml:characteristics name="Weight and Payload">
  <swe:DataRecord>
    <swe:field name="Weight">
      <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:HelicopterWeight">
        <swe:uom code="g" xlink:href=""/>
        <swe:value>1700</swe:value>
      </swe:Quantity>
    </swe:field>
    <swe:field name="Payload Capability">
      <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:PayloadCapability">
        <swe:uom code="g" xlink:href=""/>
        <swe:value>1000</swe:value>
      </swe:Quantity>
    </swe:field>
    <swe:field name="Maximum Gross Take-Off Weight">
      <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumGrossTake-OffWeight">
        <swe:uom code="g" xlink:href=""/>
        <swe:value>2700</swe:value>
      </swe:Quantity>
    </swe:field>
  </swe:DataRecord>
</sml:characteristics>
<sml:characteristics>
  sml:capabilities name="Flight">
    swe:DataRecord>
      swe:field name="Max Climb Rate">
        swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxClimbRate">
          swe:uom code="m/s" xlink:href="" />
          swe:value>2</swe:value>
      </swe:Quantity>
    </swe:field>
    swe:field name="Max Descent Rate">
        swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxDescentRate">
          swe:uom code="m/s" xlink:href="" />
          swe:value>2</swe:value>
      </swe:Quantity>
    </swe:field>
    swe:field name="Max Turn Rate">
        swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxTurnRate">
          swe:uom code="degree/second" xlink:href="" />
          swe:value>90</swe:value>
      </swe:Quantity>
    </swe:field>
    swe:field name="Approx Maximum Speed">
        swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ApproxMaximumSpeed">
          swe:uom code="km/h" xlink:href="" />
          swe:value>50</swe:value>
      </swe:Quantity>
    </swe:field>
    swe:field name="Minimum Speed">
        swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MinimumSpeed">
          swe:uom code="" xlink:href="" />
      </swe:Quantity>
    </swe:field>
  </swe:DataRecord>
</sml:capabilities>
swe:value>0</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Maximum Altitude ASL">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumAltitudeASL">
swe:uom code="m" xlink:href="" />
swe:value>2438</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Maximum Flight Time">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumFlightTime">
swe:uom code="min" xlink:href="" />
swe:value>20</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Approx Sound at 1m">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ApproxSoundat1m">
swe:uom code="dB" xlink:href="" />
swe:value>65</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Approx Sound at 3m">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ApproxSoundat3m">
swe:uom code="dB" xlink:href="" />
swe:value>60</swe:value>
</swe:Quantity>
</swe:field>
</swe:DataRecord>
</sml:capabilities>
sml:contact>
sml:ResponsibleParty>
  sml:individualName>Draganfly Innovations Inc.</sml:individualName>
  sml:contactInfo>
  sml:phone>
  sml:voice>1-800-979-9794</sml:voice>
  sml:facsimile>1-306-955-9906</sml:facsimile>
</sml:phone>
  sml:address>
    sml:deliveryPoint>Draganfly Innovations Inc. 2108 St. George Avenue Saskatoon, SK S7M0K7 Canada</sml:deliveryPoint>
  </sml:address>
</sml:contactInfo>
</sml:ResponsibleParty>
</sml:contact>
</sml:System>
</sml:member>
</sml:SensorML>

3.DRAGANFLY TANGO

<?xml version="1.0" encoding="UTF-8" ?>
<sml:SensorML rng:version="1.0.1"
  xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:ism="urn:us:gov:ic:ism:v2"
  xmlns:rng="http://relaxng.org/ns/structure/1.0"
  xmlns:sml="http://www.opengis.net/sensorML/1.0.1"
  xmlns:swe="http://www.opengis.net/swe/1.0.1"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xng="http://xng.org/1.0">
  sml:member>
    sml:System gml:id="x6">
      gml:description>Draganflyer Tango Aeroplane</gml:description>
      sml:keywords>
        sml:KeywordList>
          sml:keyword>
            UAV</sml:keyword>
        </sml:KeywordList>
      </sml:keywords>
  </sml:System>
</sml:member>
sml:keyword>Aeroplane</sml:keyword>
<sml:keyword>DRAGANFLYERTANGO</sml:keyword>

</sml:KeywordList>
</sml:keywords>
sml:identification>
sml:IdentifierList>
sml:identifier name="Short Name">
sml:Term definition="urn:ogc:def:identifier:OGC:shortName">
sml:value>Draganflyer Tango</sml:value>
</sml:Term>
</sml:identifier>
sml:identifier name="Designer Name">
sml:Term definition="urn:ogc:def:identifier:OGC:designerName">
<sml:value>Draganfly Innovations Inc. Innovative UAV Aircraft & Aerial Video Systems</sml:value>
</sml:Term>
</sml:identifier>
sml:identifier name="Builder Name">
sml:Term definition="urn:ogc:def:identifier:OGC:builderName">
sml:value>Draganfly Innovations Inc.</sml:value>
</sml:Term>
</sml:identifier>
sml:identifier name="UniqueID">
sml:Term definition="urn:ogc:def:identifier:OGC:uniqueID">
sml:value>Tango</sml:value>
</sml:Term>
</sml:identifier>
</sml:IdentifierList>
</sml:identification>

sml:classification>
sml:ClassifierList>
  sml:classifier name="Sensor Type">
    sml:Term definition="urn:ogc:classifier:sensorType">
      sml:value>Platform</sml:value>
    </sml:Term>
  </sml:classifier>
  sml:classifier name="Platform Type">
    sml:Term definition="urn:ogc:classifier:platformType">
      sml:value>Uav</sml:value>
    </sml:Term>
  </sml:classifier>
</sml:ClassifierList>
<sml:securityConstraint>
  sml:Security ism:classification="U" />
</sml:securityConstraint>
<sml:characteristics name="Aeroplane Size Dimensions">
  swe:DataRecord>
    swe:field name="Width">
      swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Width">
        swe:uom code="cm" xlink:href="" />
      swe:value>150</swe:value>
    </swe:Quantity>
  </swe:field>
    swe:field name="Length">
      swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Length">
        swe:uom code="cm" xlink:href="" />
      swe:value>120</swe:value>
    </swe:Quantity>
  </swe:field>
</sml:characteristics>
<swe:field name="Wing Area">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Wing Area">
    <swe:uom code="cm" xlink:href="" />
    <swe:value>99</swe:value>
  </swe:Quantity>
</swe:field>
</swe:DataRecord>
</sml:characteristics>

<sml:characteristics name="Weight and Payload">
  <swe:DataRecord>
    <swe:field name="Weight">
      <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Weight">
        <swe:uom code="g" xlink:href="" />
        <swe:value>2800</swe:value>
      </swe:Quantity>
    </swe:field>
    <swe:field name="Payload Capability">
      <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:PayloadCapability">
        <swe:uom code="g" xlink:href="" />
        <swe:value>1140</swe:value>
      </swe:Quantity>
    </swe:field>
    <swe:field name="Maximum Gross Take-Off Weight">
      <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumGrossTake-OffWeight">
        <swe:uom code="g" xlink:href="" />
        <swe:value>3940</swe:value>
      </swe:Quantity>
    </swe:field>
  </swe:DataRecord>
</sml:characteristics>
<sml:characteristics>
  <sml:capabilities name="Flight">
    <swe:DataRecord>
      <swe:field name="Cruise Speed">
        <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:CruiseSpeed">
          <swe:uom code="km/h" xlink:href="" />
          <swe:value>50</swe:value>
        </swe:Quantity>
      </swe:field>
      <swe:field name="Stall Speed">
        <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:StallSpeed">
          <swe:uom code="km/h" xlink:href="" />
          <swe:value>35</swe:value>
        </swe:Quantity>
      </swe:field>
      <swe:field name="Launch Type">
        <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Launch Type">
          <swe:value>Bungee Catapult</swe:value>
        </swe:Quantity>
      </swe:field>
      <swe:field name="Approx Maximum Speed">
        <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ApproxMaximumSpeed">
          <swe:uom code="km/h" xlink:href="" />
          <swe:value>95</swe:value>
        </swe:Quantity>
      </swe:field>
      <swe:field name="Minimum Speed">
        <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MinimumSpeed">
          <swe:uom code="" xlink:href="" />
          <swe:value>0</swe:value>
        </swe:Quantity>
      </swe:field>
    </swe:DataRecord>
  </sml:capabilities>
</sml:characteristics>
<swe:Quantity>
</swe:field>
swe:field name="Maximum Altitude ASL">
<swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumAltitudeASL">
swe:uom code="m" xlink:href="" />
swe:value>640</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Maximum Flight Time">
<swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumFlightTime">
swe:uom code="min" xlink:href="" />
swe:value>50</swe:value>
</swe:Quantity>
</swe:field>
</swe:DataRecord>
</sml:capabilities>
sml:contact>
sml:ResponsibleParty>
sml:individualName>Draganfly Innovations Inc.</sml:individualName>
sml:contactInfo>
sml:phone>
sml:voice>1-800-979-9794</sml:voice>
sml:facsimile>1-306-955-9906</sml:facsimile>
</sml:phone>
sml:address>
sml:deliveryPoint>Draganfly Innovations Inc. 2108 St. George Avenue Saskatoon, SK S7M0K7 Canada</sml:deliveryPoint>
</sml:address>
</sml:contactInfo>
</sml:ResponsibleParty>
4. CIRRUS 1000 BLIMP

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<sml:SensorML rng:version="1.0.1"
xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ism="urn:us:gov:ic:ism:v2"
xmlns:rn="http://relaxng.org/ns/structure/1.0"
xmlns:sml="http://www.opengis.net/sensorML/1.0.1"
xmlns:swe="http://www.opengis.net/swe/1.0.1"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xng="http://xng.org/1.0">
<sml:member>
<sml:System gml:id="cirrus 1000">
<gml:description>Cirrus 1000</gml:description>
<sml:keywords>
<sml:KeywordList>
<sml:keyword>UAV</sml:keyword>
<sml:keyword>BLIMP</sml:keyword>
<sml:keyword>CIRRUS 1000</sml:keyword>
</sml:KeywordList>
</sml:keywords>
<sml:identification>
<sml:IdentifierList>
<sml:identifier name="Short Name">
<sml:Term definition="urn:ogc:def:identifier:OGC:shortName">
<sml:value>Cirrus 1000</sml:value>
</sml:Term>
</sml:identifier>
</sml:IdentifierList>
</sml:identification>
</sml:System>
</sml:member>
</sml:SensorML>
```
<sml:classification/>
sml:securityConstraint>
sml:Security ism:classification="U" />
</sml:securityConstraint>
sml:characteristics name="Blimp Size Dimensions">
swe:DataRecord>
swe:field name="Width">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Width">
swe:uom code="cm" xlink:href="" />
swe:value>950</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Length">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Length">
swe:uom code="cm" xlink:href="" />
swe:value>250</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Top Diameter">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:TopDiameter">
swe:uom code="cm" xlink:href="" />
swe:value>250</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Height">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Height">
swe:uom code="cm" xlink:href="" />
swe:value>280</swe:value>
</swe:Quantity>
</swe:field>
<swe:DataRecord>
  <sml:characteristics name="Weight and Payload">
    <swe:DataRecord>
      <swe:field name="Weight">
        <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:HelicopterWeight">
          <swe:value>30000</swe:value>
        </swe:Quantity>
      </swe:field>
      <swe:field name="Payload Capability">
        <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:PayloadCapability">
          <swe:value>7000</swe:value>
        </swe:Quantity>
      </swe:field>
      <swe:field name="Maximum Gross Take-Off Weight">
        <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumGrossTake-OffWeight">
          <swe:value>3700</swe:value>
        </swe:Quantity>
      </swe:field>
    </swe:DataRecord>
  </sml:characteristics>
</swe:DataRecord>

<sml:capabilities name="Flight">
  <swe:DataRecord>
    <swe:field name="Envelope fabric">
      <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxClimbRate">
        <swe:value>60</swe:value>
      </swe:Quantity>
    </swe:field>
  </swe:DataRecord>
</sml:capabilities>
swe:value>2</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Max Descent Rate">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxDescentRate">
swe:uom code="m/s" xlink:href="" />
swe:value>2</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Max Turn Rate">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxTurnRate">
swe:uom code="degree/second" xlink:href="" />
swe:value>90</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Approx Maximum Speed">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ApproxMaximumSpeed">
swe:uom code="km/h" xlink:href="" />
swe:value>50</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Minimum Speed">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MinimumSpeed">
swe:uom code="" xlink:href="" />
swe:value>0</swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Maximum Altitude ASL">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumAltitudeASL">
swe:uom code="m" xlink:href="" />

69
swe:value>120 </swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Maximum Flight Time">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaximumFlightTime">
swe:uom code="min" xlink:href="" />
swe:value>20 </swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Approx Sound at 1m">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ApproxSoundat1m">
swe:uom code="dB" xlink:href="" />
swe:value>65 </swe:value>
</swe:Quantity>
</swe:field>
swe:field name="Approx Sound at 3m">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ApproxSoundat3m">
swe:uom code="dB" xlink:href="" />
swe:value>60 </swe:value>
</swe:Quantity>
</swe:field>
</swe:DataRecord>
</sml:Capabilities>
sml:Contact>
sml:ResponsibleParty>
sml:individualName>Cirrus Innovations Inc.</sml:individualName>
sml:contactInfo>
sml:phone>
sml:voice>1-800-979-9794</sml:voice>
sml:facsimile>1-306-955-9906</sml:facsimile>
CAMERA XML DATABASES
1. MICRO ANALOG

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<sml:SensorML rng:version="1.0.1"
xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
xmlns:gml="http://www.opengis.net/gml"
xmlns: rng="http://relaxng.org/ns/structure/1.0"
xmlns:sml="http://www.opengis.net/sensorML/1.0.1"
xmlns:swe="http://www.opengis.net/swe/1.0.1"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xng="http://xng.org/1.0">
  <sml:member>
    <sml:System gml:id="camera">
      <gml:name>Micro Analog</gml:name>
      <sml:keywords>
        <sml:KeywordList>
          <sml:keyword>CAMERA</sml:keyword>
          <sml:keyword>Micro Analog</sml:keyword>
          <sml:keyword>Micro Board</sml:keyword>
        </sml:KeywordList>
      </sml:keywords>
    </sml:System>
  </sml:member>
</sml:SensorML>
```
sml:Term definition="urn:ogc:classifier:platformType">
  sml:value>Platform</sml:value>
  </sml:Term>
  </sml:classifier>
  </sml:ClassifierList>
  </sml:classification>
  sml:characteristics name="Weight">
    swe:DataRecord>
      swe:field name="Camera Alone">
        swe:Quantity definition="urn:ogc:classifier:platformType">
          swe:uom code="g" xlink:href="" />
          swe:value>12</swe:value>
        </swe:Quantity>
      </swe:field>
    </swe:DataRecord>
  </sml:characteristics>
  sml:characteristics name="Features">
    swe:DataRecord>
      swe:field name="Resolution">
        swe:Quantity definition="urn:ogc:classifier:platformType">
          swe:uom code="Effective Pixels" xlink:href="" />
          swe:value>768x494</swe:value>
        </swe:Quantity>
      </swe:field>
    </swe:DataRecord>
  </sml:characteristics>
  swe:field name="Image Stabilization">
    swe:Boolean definition="urn:ogc:classifier:platformType">
      swe:value>no</swe:value>
    </swe:Boolean>
  </swe:field>
<swe:field name="Camera Speciality">
<swe:Category definition="urn:ogc:def:property:OGC:1.0:CameraSpeciality">
<swe:value>Small and Light</swe:value>
</swe:Category>
</swe:field>

<swe:field name="TV Lines Interlaced">
<swe:Quantity definition="urn:ogc:def:property:OGC:1.0:TVLinesInterlaced">
<swe:value>480</swe:value>
</swe:Quantity>
</swe:field>

<sml:characteristics name="Color">
<swe:DataRecord>
<swe:field name="Color">
<swe:Category definition="urn:ogc:def:property:OGC:1.0:Color">
<swe:value>Full Color</swe:value>
</swe:Category>
</swe:field>
</swe:DataRecord>
</sml:characteristics>

<sml:characteristics name="Includes">
<swe:DataRecord>
<swe:field name="Camera with memory card (if applicable) and all necessary cables (Note: The DMC-LX3 is modified for remote zoom, shutter and battery)">
<swe:Text />
</swe:field>
</swe:DataRecord>
</sml:characteristics>
swe:field name="Draganflyer X6 anti-vibration camera mount specifically tailored to each camera's size & weight">
swe:Text />
</swe:field>
swe:field name="Draganflyer X6 5.8 GHz wireless video transmitter, circuit board, and antenna">
swe:Text />
</swe:field>
</swe:DataRecord>
</sml:characteristics>
sml:characteristics name="Onboard Digital Video Recording Mode">
swe:DataRecord>
swe:field name="Format">
swe:Category definition="urn:ogc:def:property:OGC:1.0: FormatType">
swe:value>QuickTime</swe:value>  
</swe:Category>  
</swe:field>
swe:field name="Data speed">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Dataspeed">
swe:uom code="Mbits" xlink:href="" />
swe:value>24</swe:value>  
</swe:Quantity>  
</swe:field>
swe:field name="Framepersecond">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Framepersecond">
swe:uom code="fps" xlink:href="" />
swe:value>24</swe:value>  
</swe:Quantity>  
</swe:field>
<swe:Quantity>
</swe:Quantity>

<swe:field name="Resolution">
<swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Resolution">
<swe:uom code="pixel" xlink:href="" />
<swe:value>1280x720</swe:value>
</swe:Quantity>
</swe:field>

</swe:DataRecord>

</sml:characteristics>

<sml:capabilities name="TV Out Playback Mode">
</swe:DataRecord>

<swe:field name="NTSC">
<swe:Boolean>
<swe:value>yes</swe:value>
</swe:Boolean>
</swe:field>

<swe:field name="PAL">
<swe:Boolean>
<swe:value>yes</swe:value>
</swe:Boolean>
</swe:field>

<swe:field name="Selectable">
<swe:Boolean>
<swe:value>yes</swe:value>
</swe:Boolean>
</swe:field>

</swe:DataRecord>
</sml:capabilities>
sml:capabilities name="Zoom">
  swe:DataRecord>
  swe:field name="Optical Zoom">
    swe:Quantity definition="urn:ogc:def:property:OGC:1.0:OpticalZoom">
    swe:uom code="x" xlink:href="" />
    swe:value>2.5</swe:value>
  </swe:Quantity>
  </swe:field>
  swe:field name="Digital Zoom">
    swe:Quantity definition="urn:ogc:def:property:OGC:1.0:DigitalZoom">
    swe:uom code="x" xlink:href="" />
    swe:value>4</swe:value>
  </swe:Quantity>
  </swe:field>
  swe:field name="Controller Type">
    swe:Category definition="urn:ogc:def:property:OGC:1.0: ControllerType">
    swe:value>Remote Zoom Using Handheld Controller</swe:value>
  </swe:Category>
  </swe:field>
</swe:DataRecord>
</sml:capabilities>
sml:capabilities name="Onboard Video / Image Storage">
  swe:DataRecord>
  swe:field name="Onboard Video">
    swe:Category definition="urn:ogc:def:property:OGC:1.0:OnboardVideo">
    swe:value>SD / SDHC / MMC</swe:value>
  </swe:Category>
</swe:field>
</swe:DataRecord>
</sml:capabilities>
<swe:field name="Image Storage">
<swe:Category definition="urn:ogc:def:property:OGC:1.0:ImageStorage">
<swe:value>JPEG</swe:value>
</swe:Category>
</swe:field>
</swe:DataRecord>
</sml:capabilities>
<sml:capabilities name="Benefits">
<swe:DataRecord>
<swe:field name="Analog motion video">
<swe:Text />
</swe:field>
<swe:field name="Very low weight for longer flights">
<swe:Text />
</swe:field>
</swe:DataRecord>
</sml:capabilities>
<sml:contact>
<sml:ResponsibleParty>
<sml:individualName>Draganfly</sml:individualName>
<sml:organizationName>Draganfly Innovations Inc.</sml:organizationName>
</sml:ResponsibleParty>
</sml:contact>
<sml:parameters>
<sml:ParameterList>
<sml:parameter name="CCD / MOS / Sensorsize">
<swe:Quantity definition="urn:ogc:def:property:OGC:1.0:ccdmossensorsize">
</swe:Quantity>
</sml:parameter>
</sml:ParameterList>
</sml:parameters>
swe:uom code="" xlink:href="" />
swe:value>1/3</swe:value>
  </swe:Quantity>
</sml:parameter>
sml:parameter name="ISO / Minimum Illumination">
swe:QuantityRange definition="urn:ogc:def:property:OGC:1.0:MinimumIllumination">
swe:uom code="ISO" xlink:href="" />
swe:value>null null80 6400</swe:value>
  </swe:QuantityRange>
</sml:parameter>
sml:parameter name="Onboard Image Stabilization">
swe:Boolean definition="urn:ogc:def:property:OGC:1.0:OnboardImageStabilization">
swe:value>yes</swe:value>
  </swe:Boolean>
</sml:parameter>
sml:parameter name="ISO / Minimum Illumination">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MinimumIllumination">
swe:uom code="Lux" xlink:href="" />
swe:value>0.19</swe:value>
  </swe:Quantity>
</sml:parameter>
sml:parameter name="Lens">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Lens">
swe:uom code="mm" xlink:href="" />
swe:value>4</swe:value>
  </swe:Quantity>
</sml:parameter>
</sml:ParameterList>
2. THERMAL INFRARED

<?xml version="1.0" encoding="UTF-8" ?>

<sml:SensorML rng:version="1.0.1"
 xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
 xmlns:gml="http://www.opengis.net/gml"
 xmlns:rng="http://relaxng.org/ns/structure/1.0"
 xmlns:sml="http://www.opengis.net/sensorML/1.0.1"
 xmlns:swe="http://www.opengis.net/swe/1.0.1"
 xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xng="http://xng.org/1.0">

<sml:member>

<sml:System gml:id="camera">

<gml:name>FLIR Photon 32 FLIR Photon 640</gml:name>

<sml:keywords>

<sml:KeywordList>

<sml:keyword>CAMERA</sml:keyword>

<sml:keyword>Thermal Infrared</sml:keyword>

<sml:keyword>FLIR Photon</sml:keyword>

</sml:KeywordList>

</sml:keywords>

<sml:identification>

<sml:IdentifierList>

<sml:identifier name="Short Name">

<sml:Term definition="urn:ogc:def:identifier:OGC:shortName">

<sml:value>FLIR Photon</sml:value>

</sml:Term>

</sml:identifier>

</sml:IdentifierList>

</sml:identification>

</sml:System>

</sml:member>

</sml:SensorML>
<sml:Term>
  <sml:identifier>
    <sml:identifier name="Long Name">
      <sml:Term definition="urn:ogc:def:identifier:OGC:longName">
        <sml:value>FLIR Photon 320 FLIR Photon 640</sml:value>
      </sml:Term>
    </sml:identifier>
  </sml:identifier>
</sml:Term>

<sml:Term>
  <sml:identifier>
    <sml:identifier name="Manufacturer Name">
      <sml:Term definition="urn:ogc:def:identifier:OGC:manufacturerName">
        <sml:value>FLIR</sml:value>
      </sml:Term>
    </sml:identifier>
  </sml:identifier>
</sml:Term>

<sml:IdentifierList>
  <sml:identification>
    <sml:classification>
      <sml:ClassifierList>
        <sml:classifier name="Sensor Type">
          <sml:Term definition="urn:ogc:classifier:sensorType">
            <sml:value>Platform</sml:value>
          </sml:Term>
        </sml:classifier>
        <sml:classifier name="Platform Type">
          <sml:Term definition="urn:ogc:classifier:platformType">
            <sml:value>Camera</sml:value>
          </sml:Term>
        </sml:classifier>
      </sml:ClassifierList>
    </sml:classification>
  </sml:identification>
</sml:IdentifierList>
sml:characteristics name="Weight">
  swe:DataRecord>
    swe:field name="photon 320 Weight">
      swe:DataRecord>
        swe:field name="Camera Alone">
          swe:Quantity definition="urn:ogc:def:property:OGC:1.0:cameraalone">
            swe:uom code="g" xlink:href="" />
            swe:value>153</swe:value>
            </swe:Quantity>
        </swe:field>
      </swe:DataRecord>
    </swe:field>
  </swe:DataRecord>
  swe:field name="photon 640 Weight">
    swe:DataRecord>
      swe:field name="Camera Alone">
        swe:Quantity definition="urn:ogc:def:property:OGC:1.0:cameraalone">
          swe:uom code="g" xlink:href="" />
          swe:value>227</swe:value>
          </swe:Quantity>
        </swe:field>
      </swe:DataRecord>
    </swe:field>
  </swe:DataRecord>
  swe:field name="Camera & Mount">
    swe:Quantity definition="urn:ogc:def:property:OGC:1.0:cameraandmountweight">
      swe:uom code="g" xlink:href="" />
      swe:value>228</swe:value>
      </swe:Quantity>
    </swe:field>
  </swe:DataRecord>
</sml:characteristics>
<swe:field name="Features for photon 320">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Resolution">
    <swe:uom code="Effective Pixels" xlink:href="" />
    <swe:value>324x256</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="LENS">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:lens">
    <swe:uom code="mm" xlink:href="" />
    <swe:value>14.25</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="TV Lines Interlaced">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:TVLinesInterlaced">
    <swe:uom code="null" xlink:href="" />
    <swe:value>240</swe:value>
  </swe:Quantity>
</swe:field>
swe:field name="Features for photon 320">
swe:DataRecord>
swe:field name="Resolution">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Resolution">
swe:uom code="Effective Pixels" xlink:href="" />
swe:value>640x512</swe:value>
  </swe:Quantity>
</swe:field>
swe:field name="LENS">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:lens">
swe:uom code="mm" xlink:href="" />
swe:value>21.5</swe:value>
  </swe:Quantity>
</swe:field>
swe:field name="TV Lines Interlaced">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:TVLinesInterlaced">
swe:uom code="" xlink:href="" />
swe:value>480</swe:value>
  </swe:Quantity>
</swe:field>
</swe:DataRecord>
</swe:field>
swe:field name="Both (640 and 320)">
swe:DataRecord>
swe:field name="Sensor Type">
  swe:Category definition="urn:ogc:def:property:OGC:1.0:SensorType">
  swe:value>Vox Microbolometer</swe:value>
</swe:Category>
</swe:field>

swe:field name="Thermal Imager">
  swe:Category definition="urn:ogc:def:property:OGC:1.0:ThermalImager">
  swe:value>Uncooled LWIR Thermal Imager</swe:value>
</swe:Category>
</swe:field>

swe:field name="Infrared Type">
  swe:Category definition="urn:ogc:def:property:OGC:1.0:InfraredType">
  swe:value>Forward Looking</swe:value>
</swe:Category>
</swe:field>

swe:field name="Spectral Band">
  swe:QuantityRange definition="urn:ogc:def:property:OGC:1.0:SpectralBand">
  swe:uom code="μm" xlink:href="" />
  swe:value>null null7.5 13.5</swe:value>
</swe:QuantityRange>
</swe:field>
</swe:DataRecord>

</swe:field>
</swe:DataRecord>
</sml:characteristics>

sml:characteristics name="Color">
  swe:DataRecord>
  swe:field name="Color"
swe:Category definition="urn:ogc:def:property:OGC:1.0:Color">
swe:value>Full Color</swe:value>
</swe:Category>  
</swe:field>  
swe:field name="Polarity Control">  
swe:Boolean definition="urn:ogc:def:property:OGC:1.0:PolarityControl">
swe:value>yes</swe:value>
</swe:Boolean>  
</swe:field>  
swe:field name="Color Option">  
swe:Category definition="urn:ogc:def:property:OGC:1.0:ColorOption">
swe:value>False Color Option</swe:value>
</swe:Category>  
</swe:field>  
</sml:characteristics>  
sml:characteristics name="Includes">  
swe:DataRecord>

swe:field name="Camera with memory card (if applicable) and all necessary cables">  
swe:Text />  
</swe:field>  
swe:field name="Draganflyer X6 anti-vibration camera mount specifically tailored to each camera's size & weight">  
swe:Text />  
</swe:field>  
swe:field name="Draganflyer X6 5.8 GHz wireless video transmitter, circuit board, and antenna">  
swe:Text />  
</swe:field>
<sml:characteristics>
  <sml:capabilities name="TV Out Playback Mode">
    <swe:DataRecord>
      <swe:field name="NTSC">
        <swe:Boolean>
          <swe:value>y</swe:value>
        </swe:Boolean>
      </swe:field>
      <swe:field name="PAL">
        <swe:Boolean>
          <swe:value>y</swe:value>
        </swe:Boolean>
      </swe:field>
      <swe:field name="Selectable">
        <swe:Boolean>
          <swe:value>y</swe:value>
        </swe:Boolean>
      </swe:field>
    </swe:DataRecord>
  </sml:capabilities>
  <sml:capabilities name="Zoom">
    <swe:DataRecord>
      <swe:field name="320 photon Zoom">
        <swe:DataRecord>
          <swe:field name="Digital Zoom">
            <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:DigitalZoom">
            </swe:Quantity>
          </swe:field>
        </swe:DataRecord>
      </swe:field>
    </swe:DataRecord>
  </sml:capabilities>
</sml:characteristics>
swe:field name="640 photon Zoom">
  swe:DataRecord>
  swe:field name="Digital Zoom">
  swe:Quantity definition="urn:ogc:def:property:OGC:1.0:DigitalZoom">
  swe:uom code="x" xlink:href="" />
  swe:value>4 and 2</swe:value>
  </swe:Quantity>
  </swe:field>
  </swe:DataRecord>
  </swe:field>
</sml:capabilities>
sml:capabilities name="Onboard Video / Image Storage">
  swe:DataRecord>
  swe:field name="Onboard Video">
  swe:Category definition="urn:ogc:def:property:OGC:1.0:OnboardVideo">
  swe:value>none</swe:value>
  </swe:Category>
  </swe:field>
  swe:field name="Image Storage">
  swe:Category definition="urn:ogc:def:property:OGC:1.0:ImageStorage">
  swe:value>none(wireless transmits only)</swe:value>
<swe:Category/>
</swe:field>
</swe:DataRecord>
</sml:capabilities>
sml:capabilities name="Benefits">
swe:DataRecord>
swe:field name="Thermal infrared motion video">
swe:Text />
</swe:field>
</swe:DataRecord>
swe:field name="Heat vision">
swe:Text />
</swe:field>
</swe:DataRecord>
swe:field name="Search and Rescue">
swe:Text />
</swe:field>
</swe:DataRecord>
</sml:capabilities>
sml:contact>
sml:ResponsibleParty>
sml:individualName>Draganfly</sml:individualName>
sml:organizationName>Draganfly Innovations Inc.</sml:organizationName>
</sml:ResponsibleParty>
</sml:contact>
sml:interfaces>
sml:InterfaceList />
</sml:interfaces>
sml:parameters>
sml:ParameterList>
}
<sml:parameter name="Onboard Image Stabilization">
  <swe:Boolean definition="urn:ogc:def:property:OGC:1.0:OnboardImageStabilization">
    <swe:value>no</swe:value>
  </swe:Boolean>
</sml:parameter>

<sml:parameter name="320 Focal Length">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:FocalLength">
    <swe:uom code="mm" xlink:href=""/>
    <swe:value>14.25</swe:value>
  </swe:Quantity>
</sml:parameter>

<sml:parameter name="640 Focal Length">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:FocalLength">
    <swe:uom code="mm" xlink:href=""/>
    <swe:value>21.5</swe:value>
  </swe:Quantity>
</sml:parameter>

<sml:parameter name="320 ISO / Minimum Illumination">
  <swe:ConditionalData definition="urn:ogc:def:property:OGC:1.0:MinimumIllumination"/>
</sml:parameter>

<sml:parameter name="640 ISO / Minimum Illumination">
  <swe:ConditionalData definition="urn:ogc:def:property:OGC:1.0:MinimumIllumination"/>
</sml:parameter>

<sml:components>
  <sml:ComponentList>
    <sml:component name="CCD / MOS /Sensor">
    </sml:component>
  </sml:ComponentList>
</sml:components>
<sml:System>
  <gml:name>Microbolometer</gml:name>
  <sml:identification>
    <sml:IdentifierList>
      <sml:identifier name="Short Name">
        <sml:Term definition="urn:ogc:def:identifier:OGC:shortName">
          <sml:value>Vox Microbolometer</sml:value>
        </sml:Term>
      </sml:identifier>
    </sml:IdentifierList>
  </sml:identification>
  <sml:characteristics name="Spectral Band">
    <swe:DataRecord>
      <swe:field name="Spectral Band">
        <swe:QuantityRange definition="urn:ogc:def:property:OGC:1.0:SpectralBand">
          <swe:uom code="μm" xlink:href="" />
          <swe:value>null null7.5 13.5</swe:value>
        </swe:QuantityRange>
      </swe:field>
    </swe:DataRecord>
  </sml:characteristics>
</sml:System>
3. LOW LIGHT

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<sml:SensorML xmlns:a="http://relaxng.org/ns/compatibility/annotations/1.0"
xmlns:gml="http://www.opengis.net/gml"
xmlns: rng="http://relaxng.org/ns/structure/1.0"
xmlns:sml="http://www.opengis.net/sensorML/1.0.1"
xmlns:swe="http://www.opengis.net/swe/1.0.1"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xng="http://xng.org/1.0">
<sml:member>
<sml:System gml:id="camera">
<gml:name>Watec WAT-902H2 Ultimate</gml:name>
<sml:keywords>
<sml:KeywordList>
<sml:keyword>Low Light</sml:keyword>
<sml:keyword>Camera</sml:keyword>
<sml:keyword>Watec</sml:keyword>
<sml:keyword>WAT-902H2</sml:keyword>
</sml:KeywordList>
</sml:keywords>
<sml:identification>
<sml:IdentifierList>
<sml:identifier name="Short Name">
<sml:Term definition="urn:ogc:def:identifier:OGC:shortName">
<sml:value>Watec Ultimate</sml:value>
</sml:Term>
</sml:identifier>
<sml:identifier name="Long Name">
<sml:Term definition="urn:ogc:def:identifier:OGC:longName">
</sml:Term>
</sml:identifier>
</sml:IdentifierList>
</sml:identification>
</sml:System>
</sml:member>
</sml:SensorML>
```
Camera Alone

Camera & Mount

Features

Resolution

CCD
<swe:Quantity gml:id="urn:ogc:def:property:OGC:1.0:TVLinesInterlaced">
  <swe:uom code="" xlink:href="" />
  <swe:value>1/2</swe:value>
</swe:Quantity>
<swe:field name="TV Lines Interlaced">
  <swe:Quantity gml:id="urn:ogc:def:property:OGC:1.0:TVLinesInterlaced">
    <swe:uom code="" xlink:href="" />
    <swe:value>570</swe:value>
  </swe:Quantity>
</swe:field>
<swe:field name="Minimum llumination">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Minimum llumination">
    <swe:uom code="Lux" xlink:href="" />
    <swe:value>0.0001</swe:value>
  </swe:Quantity>
</swe:field>
</swe:DataRecord>
</sml:characteristics>
<sml:characteristics name="Color">
  <swe:DataRecord>
    <swe:field name="Color">
    </swe:field>
    <swe:field name="Color">
      <swe:Category definition="urn:ogc:def:property:OGC:1.0:Color">
        <swe:value>Monochrome</swe:value>
      </swe:Category>
    </swe:field>
  </swe:DataRecord>
</sml:characteristics>
sml:characteristics name="Includes">
swe:DataRecord>
swe:field name="Camera with memory card (if applicable) and all necessary cables">
swe:Text />
</swe:field>
swe:field name="Draganflyer X6 anti-vibration camera mount specifically tailored to each camera's size & weight">
swe:Text />
</swe:field>
swe:field name="Draganflyer X6 5.8 GHz wireless video transmitter, circuit board, and antenna">
swe:Text />
</swe:field>
swe:DataRecord>
</sml:characteristics>
sml:capabilities name="TV Out Playback Mode">
swe:DataRecord>
swe:field name="TV Out Playback Mode">
swe:DataRecord />
</swe:field>
swe:field name="NTSC">
swe:Boolean>
swe:value>yes</swe:value>
</swe:Boolean>
</swe:field>
swe:field name="PAL"/>
swe:Boolean>
swe:value>yes</swe:value>
</swe:Boolean>
</swe:field>
swe:field name="Selectable">
swe:Boolean>
swe:value>no</swe:value>
</swe:Boolean>
</swe:field>
</swe:DataRecord>
</sml:capabilities>
sml:capabilities name="Zoom">
swe:DataRecord>
swe:field name="Zoom">
swe:DataRecord />
</swe:field>
swe:field name="Optical Zoom">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:OpticalZoom">
swe:uom code="x" xlink:href="" />
swe:value>3</swe:value>
</swe:Quantity>
</swe:field>
</swe:DataRecord>
</sml:capabilities>
sml:capabilities name="Onboard Video / Image Storage">
swe:DataRecord>
swe:field name="Onboard Video / Image Storage">
swe:DataRecord />

Onboard Video

Image Storage

Benefits

Low light motion video

Clear video where other cameras would only show black
Draganfly Innovations Inc.

<sml:parameters>
  <sml:ParameterList>
    <sml:parameter name="Focal Length">
      <swe:QuantityRange definition="urn:ogc:def:property:OGC:1.0:FocalLength">
        <swe:uom code="mm" xlink:href="" />
        <swe:value>4 12</swe:value>
      </swe:QuantityRange>
    </sml:parameter>
    <sml:parameter name="Onboard Image Stabilization">
      <swe:Boolean definition="urn:ogc:def:property:OGC:1.0:OnboardImageStabilization">
        <swe:value>no</swe:value>
      </swe:Boolean>
    </sml:parameter>
    <sml:parameter name="ISO / Minimum Illumination">
      <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MinimumIllumination">
        <swe:uom code="Lux" xlink:href="" />
        <swe:value>0.0001</swe:value>
      </swe:Quantity>
    </sml:parameter>
  </sml:ParameterList>
</sml:parameters>
Panasonic has today announced the Lumix DMC-LX3. The camera is aimed at DSLR users who are looking for a compact camera to complement their existing SLR gear. Consequently the LX3 comes with comprehensive manual controls and a fast F2.0-F2.8 24-60mm (35mm equivalent) Leica DC Vario-Summicron lens. Images are captured on a 1/1.63-inch CCD sensor sporting 10.1 million effective pixels.
sml:Term
sml:identifier
sml:identifier name="Long Name">
sml:Term definition="urn:ogc:def:identifier:OGC:longName">
sml:value>Panasonic Lumix DMC-LX3</sml:value>
  </sml:Term>
  </sml:identifier>
  </sml:IdentifierList>
  </sml:identification>
sml:classification>
sml:ClassifierList>
sml:classifier name="Sensor Type">
sml:Term definition="urn:ogc:classifier:sensorType">
sml:value>Platform</sml:value>
  </sml:Term>
  </sml:classifier>
  </sml:ClassifierList>
  </sml:classification>

sml:identifier name="Manufacturer Name">
sml:Term definition="urn:ogc:def:identifier:OGC:manufacturerName">
sml:value>Panasonic</sml:value>
  </sml:Term>
  </sml:identifier>

sml:classifier name="Platform Type">
sml:Term definition="urn:ogc:classifier:platformType">
sml:value>Camera</sml:value>
  </sml:Term>
  </sml:classifier>
  </sml:ClassifierList>
  </sml:classification>
swe:field name="Camera Alone">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:cameraalone">
swe:uom code="g" xlink:href="" />
swe:value>227</swe:value>
</swe:field>

swe:field name="Camera & Mount">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:cameraandmountweight">
swe:uom code="g" xlink:href="" />
swe:value>302</swe:value>
</swe:field>
</sml:characteristics>

sml:characteristics name="Features">
swe:field name="Resolution">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Resolution">
swe:uom code="Megapixel" xlink:href="" />
swe:value>10.1</swe:value>
</swe:field>

swe:field name="Wide-Angle LENS">
swe:Quantity definition="urn:ogc:def:property:OGC:1.0:wideanglelens">
swe:uom code="mm" xlink:href="" />
swe:value>24</swe:value>
<swe:field name="Optical Zoom">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:OpticalZoom">
    <swe:uom code="x" xlink:href="" />
    <swe:value>2.5</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="Max Image Size">
  <swe:Quantity definition="urn:ogc:def:property:OGC:1.0:MaxImageSize">
    <swe:uom code="null" xlink:href="" />
    <swe:value>3648x2736</swe:value>
  </swe:Quantity>
</swe:field>

<swe:field name="Image Stabilization">
  <swe:Boolean definition="urn:ogc:def:property:OGC:1.0:ImageStabilization">
    <swe:value>yes</swe:value>
  </swe:Boolean>
</swe:field>

<swe:field name="Save place">
  <swe:Category gml:id="urn:ogc:def:property:OGC:1.0:SavePlace">
    <swe:value>SD/SDHC/MMC Memory Card</swe:value>
  </swe:Category>
</swe:field>

<swe:DataRecord>
<swe:field name="Color">
  <swe:Category gml:id="urn:ogc:def:property:OGC:1.0:Color">
    <swe:value>Full Color</swe:value>
  </swe:Category>
</swe:field>

<sml:characteristics name="Includes">
  <swe:field name="Camera with memory card (if applicable) and all necessary cables (Note: The DMC-LX3 is modified for remote zoom, shutter and battery)">
    <swe:Text />
  </swe:field>
</sml:characteristics>

<sml:characteristics name="Onboard Digital Video Recording Mode">
  <swe:DataRecord>
    <swe:field name="Format">
      <swe:Category definition="urn:ogc:def:property:OGC:1.0:">
        <swe:value>QuickTime</swe:value>
      </swe:Category>
    </swe:field>
  </swe:DataRecord>
</sml:characteristics>
swe:field name="Data speed">
<swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Dataspeed">
<swe:uom code="" xlink:href="" />
<swe:value>24</swe:value>
</swe:Quantity>
</swe:field>

swe:field name="Framepersecond">
<swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Framepersecond">
<swe:uom code="" xlink:href="" />
<swe:value>24</swe:value>
</swe:Quantity>
</swe:field>

swe:field name="Resolution">
<swe:Quantity definition="urn:ogc:def:property:OGC:1.0:Resolution">
<swe:value>1280x720</swe:value>
</swe:Quantity>
</swe:field>
</swe:DataRecord>
</sml:characteristics>

sml:capabilities name="TV Out Playback Mode">
<swe:DataRecord>
<swe:field name="NTSC">
<swe:Boolean>
<swe:value>yes</swe:value>
</swe:Boolean>
</swe:field>
<swe:field name="PAL">
swe:Boolean>
  swe:value>yes</swe:value>
  </swe:Boolean>
  </swe:field>
</swe:DataRecord>
</sml:capabilities>
sml:capabilities name="Zoom">
  swe:DataRecord>
  swe:field name="Optical Zoom">
    swe:Quantity definition="urn:ogc:def:property:OGC:1.0:OpticalZoom">
    swe:uom code="x" xlink:href="" />
    swe:value>2.5</swe:value>
    </swe:Quantity>
    </swe:field>
  swe:field name="Digital Zoom">
    swe:Quantity gml:id="urn:ogc:def:property:OGC:1.0:OpticalZoom">
    swe:uom code="x" xlink:href="" />
    swe:value>4</swe:value>
    </swe:Quantity>
    </swe:field>
  swe:field name="Controller">
    swe:Category gml:id="urn:ogc:def:property:OGC:1.0:Controller">
    swe:value>Remote Zoom Using Handheld Controller</swe:value>
  </swe:Category>
</swe:DataRecord>
</sml:capabilities>
swe:Category definition="urn:ogc:def:property:OGC:1.0:OnboardVideo">
swe:value>SD / SDHC / MMC</swe:value>
</swe:Category>
</swe:field>

swe:field name="Image Storage">
swe:Category definition="urn:ogc:def:property:OGC:1.0:ImageStorage">
swe:value>JPEG</swe:value>
</swe:Category>
</swe:field>
</swe:DataRecord>
</sml:capabilities>

sml:capabilities name="Benefits">
swe:DataRecord>

swe:field name="Professional quality digital still photographs">
swe:Text /
</swe:field>

swe:field name="Aerial mapping">
swe:Text /
</swe:field>

swe:field name="Real Estate photography">
swe:Text /
</swe:field>
<swe:field name="Remote zoom, tilt and shutter">
<swe:Text />
</swe:field>
</swe:DataRecord>
</sml:capabilities>
<sml:contact>
<sml:ResponsibleParty>
<sml:individualName>Draganfly</sml:individualName>
<sml:organizationName>Draganfly Innovations Inc.</sml:organizationName>
<sml:contactInfo>
<sml:phone>
<sml:facsimile>306-955-9907</sml:facsimile>
</sml:phone>
<sml:address>
<sml:city />
<sml:administrativeArea />
<sml:postalCode />
<sml:country />
<sml:electronicMailAddress>info@draganfly.com</sml:electronicMailAddress>
</sml:address>
</sml:contactInfo>
</sml:ResponsibleParty>
</sml:contact>
<sml:parameters>
<sml:ParameterList>
<sml:parameter name="Focal Length">
<swe:QuantityRange definition="urn:ogc:def:property:OGC:1.0:FocalLength"/>
</sml:parameter>
</sml:ParameterList>
</sml:parameters>
<sml:ParameterList>
  <sml:parameters>
    <sml:parameter name="CCD / MOS / Sensor">
      <sml:parameter name="swe:Quantity">
        <swe:QuantityRange>
          <swe:value>5.1 12.8</swe:value>
        </swe:QuantityRange>
      </sml:parameter>
      <sml:parameter name="swe:Quantity">
        <swe:QuantityRange>
          <swe:value>1/1.63</swe:value>
        </swe:QuantityRange>
      </sml:parameter>
      <sml:parameter name="ISO / Minimum Illumination">
        <swe:QuantityRange>
          <swe:value>null 0 80 6400</swe:value>
        </swe:QuantityRange>
      </sml:parameter>
      <sml:parameter name="Onboard Image Stabilization">
        <swe:Boolean>
          <swe:value>yes</swe:value>
        </swe:Boolean>
      </sml:parameter>
    </sml:parameters>
  </sml:ParameterList>
</sml:systems>

<sml:components>
  <sml:ComponentList>
    <sml:component name="Leica DC Vario-Summicron">
      <sml:System>
        <sml:characteristics name="Lens">
          <swe:uom code="mm" xlink:href="" />
          <swe:value>null 5.1 12.8</swe:value>
        </swe:QuantityRange>
        <swe:uom code="inch CCD" xlink:href="" />
        <swe:value>1/1.63</swe:value>
      </swe:Quantity>
    </sml:component>
  </sml:ComponentList>
</sml:components>
swe:DataRecord>
  swe:field name="Lens">
  swe:DataRecord />
  </swe:field>
  swe:field name="Wide-Angle Lens">
  swe:QuantityRange>
  swe:uom code="mm" xlink:href="" />
  swe:value>null null24 60</swe:value>
  </swe:QuantityRange>
  </swe:field>
  </swe:DataRecord>
</sml:characteristics>
</sml:System>
</sml:component>
</sml:ComponentList>
</sml:components>
</sml:System>
</sml:member>
</sml:SensorML>

**CODE FOR UAV QUESTIONNAIRE**

```php
<?php
```
'urn:ogc:def:property:OGC:1.0:PayloadCapability',
'urn:ogc:def:property:OGC:1.0:MaximumGrossTake-OffWeight',
'urn:ogc:def:property:OGC:1.0:MaxClimbRate',
'urn:ogc:def:property:OGC:1.0:MaxDescentRate',
'urn:ogc:def:property:OGC:1.0:MaxTurnRate',
'urn:ogc:def:property:OGC:1.0:ApproxMaximumSpeed',
'urn:ogc:def:property:OGC:1.0:MaximumAltitudeASL'
'urn:ogc:def:property:OGC:1.0:MaximumFlightTime',
'urn:ogc:def:property:OGC:1.0:ApproxSoundat1m',
'urn:ogc:def:property:OGC:1.0:ApproxSoundat3m');
// questions will be searched separately with the aim of finding values

$questions = array('question1' => 'list1', 'question2' => 'list2',
'question3' => 'list3', 'question4' => 'list4', 'question5' => 'list5',
'question6' => 'list6', 'question7' => 'list7', 'question8' => 'list8',
'question9' => 'list9', 'question10' => 'list10', 'question11' => 'list11',
'question12' => 'list12', 'question13' => 'list13', 'question14' => 'list14',
'question15' => 'list15');
// questions wanted to be in an array for handling job with less command

$cnumber_draganflyx6=0;
$cnumber_draganflyx8=0;
$cnumber_draganflytango=0;
$cnumber_cirrus_blimp=0;
// counters for all UAV biggest value will be the most suitable UAV

if(isset($_POST['submit'])) {
    foreach($questions as $questionkey => $questionvalue) {
        $questionpost = isset($_POST[$questionvalue]) ? $_POST[$questionvalue] : '';
        if($questionkey == empty($questionpost)) {
            $errormsg = 'Please select an answer for '. $questionkey;
            echo $errormsg . '<br />';
        } } }
// If an question is not checked this error message will be shown
foreach ($questions as $questionkey => $questionvalue) {
    foreach ($search_questions as $search_text) {
        $sml:SensorML = new SimpleXMLElement('draganflyx6.xml', 'null', 'true'); // all xml files wanted to be separate its elements
        $quantity_characteristics = $sml:SensorML->xpath('sml:member/sml:System/sml:characteristics/swe:DataRecord/swe:field/swe:Quantity');
        foreach ($quantity_characteristics as $swe:Quantity) {
            if ($swe:Quantity ['definition'] == $search_text && $swe:Quantity->swe:value == $_POST[$questionvalue])
                $counter_draganflyx6++; } // if search text and question text is available in characteristics specifications increase counter

        foreach ($quantity_capabilities as $swe:Quantity) {
            if ($swe:Quantity ['definition'] == $search_text && $swe:Quantity->swe:value == $_POST[$questionvalue])
                $counter_draganflyx6++; } // if search text and question text is available in characteristics specifications increase counter
    }
}

draganflyx8.xml

draganflyx8.xml

draganflyx8.xml

foreach ($questions as $questionkey => $questionvalue) {
forecah ($search_questions as $search_text) {
$sml:SensorML= new SimpleXMLElement('draganflyertango.xml', null, true); //all xml files wanted to be seperate its elements
$quantity_characteristics= $sml:SensorML-
>xpath('sml:member/sml:System/sml:characteristics/swe:DataRecord/swe:field/swe:Quantity');
foreach ( $quantity_characteristics as $swe:Quantity ) {
if ($swe:Quantity['definition']== $search_text && $swe:Quantity->swe:value== $_POST[$questionvalue])
$counter_dragenflyer_tango++; }   // if search text and question text is avaible in characteristics specifications
increase counter

$quantity_capabilities= $sml:SensorML-
>xpath('sml:member/sml:System/sml:capabilities/swe:DataRecord/swe:field/swe:Quantity');
foreach ( $quantity_capabilities as $swe:Quantity ) {
if ($swe:Quantity['definition']== $search_text && $swe:Quantity->swe:value== $_POST[$questionvalue])
$counter_dragenflyer_tango++; }    // if search text and question text is avaible in characteristics specifications
increase counter

foreach ($questions as $questionkey => $questionvalue) {
forecah ($search_questions as $search_text) {
$sml:SensorML= new SimpleXMLElement('cirrusblimp.xml', null, true); //all xml files wanted to be seperate its elements
$quantity_characteristics= $sml:SensorML-
>xpath('sml:member/sml:System/sml:characteristics/swe:DataRecord/swe:field/swe:Quantity');
foreach ( $quantity_characteristics as $swe:Quantity ) {
if ($swe:Quantity['definition']== $search_text && $swe:Quantity->swe:value== $_POST[$questionvalue])
$counter_cirrus_blimp++; }   // if search text and question text is avaible in characteristics specifications
increase counter

$quantity_capabilities= $sml:SensorML-
>xpath('sml:member/sml:System/sml:capabilities/swe:DataRecord/swe:field/swe:Quantity');
foreach ( $quantity_capabilities as $swe:Quantity ) {
if ($swe:Quantity['definition']== $search_text && $swe:Quantity->swe:value== $_POST[$questionvalue])
$counter_cirrus_blimp++; }    // if search text and question text is avaible in characteristics specifications
increase counter

foreach ($questions as $questionkey => $questionvalue) {
forecah ($search_questions as $search_text) {
$sml:SensorML= new SimpleXMLElement('cirrusblimp.xml', null, true); //all xml files wanted to be seperate its elements
$quantity_characteristics= $sml:SensorML-
>xpath('sml:member/sml:System/sml:characteristics/swe:DataRecord/swe:field/swe:Quantity');
foreach ( $quantity_characteristics as $swe:Quantity ) {
if ($swe:Quantity['definition']== $search_text && $swe:Quantity->swe:value== $_POST[$questionvalue])
$counter_cirrus_blimp++; }   // if search text and question text is avaible in characteristics specifications
increase counter

$quantity_capabilities= $sml:SensorML-
>xpath('sml:member/sml:System/sml:capabilities/swe:DataRecord/swe:field/swe:Quantity');}
foreach ($quantity_capabilities as $swe:Quantity) {
    if ($swe:Quantity['definition'] == $search_text && $swe:Quantity['swe:value'] == $_POST['questionvalue'])
        $counter_cirrus_blimp++;  } // if search text and question text is available in characteristics specifications increase counter
}

$result = max($counter.draganflyx6, $counter.draganflyx8, $counter.draganflyer_tango, $counter.cirrus_blimp);
if ($result == $counter.draganflyx6)
    echo " DraganflyX6 is chosen as most suitable UAV for that condition";
if ($result == $counter.draganflyx8)
    echo " DraganflyX8 is chosen as most suitable UAV for that condition";
if ($result == $counter.draganflyer_tango)
    echo " Draganflyer Tango is chosen as most suitable UAV for that condition";
if ($result == $counter.cirrus_blimp)
    echo " Cirrus blimp is chosen as most suitable UAV for that condition";
?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
<head>
<title>Questionnaire</title>
</head>
<body>
<form name="SENSING MISSION MANAGEMENT PROGRAM(UAV)" method="post" action="../../Documents/Unnamed Site 7/uav.php">
<p>SENSING MISSION MANAGEMENT PROGRAM(UAV)</p>
<table width="611" border="0" bordercolor="#FFFFFF">
<tr>
<td width="170"> <br />
1 . Width<br />
<input type="radio" name="list1" value="87" /> Narrow<br />
<input type="radio" name="list1" value="91" /> Normal<br />
</td>
</tr>
</table>
</form>
</body>
</html>
<table>
<thead>
<tr>
<th>Feature</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Width</td>
<td>Wide, Extra-Wide</td>
</tr>
<tr>
<td>Length</td>
<td>Short, Normal, Long, Long</td>
</tr>
<tr>
<td>Top Diameter</td>
<td>Low, High</td>
</tr>
<tr>
<td>Height</td>
<td>Small, Big</td>
</tr>
<tr>
<td>Weight</td>
<td>Light, Normal, Heavy, Heaviest</td>
</tr>
<tr>
<td>Payload Capability</td>
<td>Low, Normal, High, Highest</td>
</tr>
<tr>
<td>Maximum Gross Take-Off Weight</td>
<td>Light, Normal, Heavy, Heaviest</td>
</tr>
<tr>
<td>Max Climb Rate</td>
<td>Low, High</td>
</tr>
<tr>
<td>Max Descent Rate</td>
<td>Low, High</td>
</tr>
<tr>
<td>Max Turn Rate</td>
<td>Low, High</td>
</tr>
<tr>
<td>Approx Maximum Speed</td>
<td></td>
</tr>
</tbody>
</table>
116

<input type="radio" name="list11" value="50" /> Slow<br />
<input type="radio" name="list11" value="300" /> Fast<br /><br />
12. Maximum Altitude ASL<br />
<input type="radio" name="list12" value="120" /> Low<br />
<input type="radio" name="list12" value="640" /> Normal<br />
<input type="radio" name="list12" value="2438" /> High<br />

13. Maximum Flight Time<br />
<input type="radio" name="list13" value="20" /> Short<br />
<input type="radio" name="list13" value="50" /> Long<br /><br />
14. Approx Sound at 1m<br />
<input type="radio" name="list14" value="1" /> No sound<br />
<input type="radio" name="list14" value="60" /> Noisly<br /><br />
15. Approx Sound at 3m<br />
<input type="radio" name="list15" value="1" /> No sound<br />
<input type="radio" name="list15" value="65" /> Noisly<br /><br />
</td>
</tr>
</table>
<p>
<label></label>
<input type="submit" name="Submit" value="Submit" />
<br>
</p>
</form>
</body>
</html>

CODE CAMERA UAV QUESTIONAIRE

<?php
$search_questions=array('Features','Benefits','CCD / MOS/Sensor','Weight','Max Pixels','Color','Onboard Digital Video Recording Mode','TV Out Playback Mode','Onboard Video/ Image Storage','Zoom','Lens','Focal Length','Onboard Image Stabilization','ISO / Minimum Illumination','Type of data');
// questions will be searched seperately with the aim of finding values
$questions = array('question1' => 'list1', 'question2' => 'list2',
'question3' => 'list3', 'question4' => 'list4', 'question5' => 'list5',
'question6' => 'list6', 'question7' => 'list7', 'question8' => 'list8',
'question9' => 'list9', 'question10' => 'list10', 'question11' => 'list11',
'question12' => 'list12', 'question13' => 'list13', 'question14' => 'list14',
'question15' => 'list15');
//questions wanted to be in an array for handling job with less command

$counter_Low_Light=0;
$counter_HD_Video=0;
$counter_Thermal_Infrared=0;
$counter_Micro_Analog=0;
//counters for all UAV biggest value will be the most suitable UAV

if(isset($_POST['submit'])) {
    foreach($questions as $questionkey => $questionvalue) {
        $questionpost = isset($_POST[$questionvalue]) ? $_POST[$questionvalue] : '';
        if($questionkey == empty($questionpost)) {
            $errormessage = 'Please select an answer for '. $questionkey;
            echo $errormessage . '<br />';
        }
    }
} //if an question is not checked this error message will be shown

foreach($questions as $questionkey => $questionvalue) {
    $Low_Light= file_get_contents("Lowlight.xml");
    if( !empty( preg_match($_POST[$questionvalue], NULL , $Low_Light))
        && empty( preg_match($], NULL , $Low_Light)) )
        $counter_Low_Light++;
}

foreach($questions as $questionkey => $questionvalue) {
$HD_Video= file_get_contents("HDVideo.xml");
if( !empty( preg_match($_POST[$questionvalue], NULL , $HD_Video)))
$counter_HD_Video++;
}}}

foreach($questions as $questionkey => $questionvalue) {
$Thermal_Infrared= file_get_contents("ThermalInfrared.xml");
if( !empty( preg_match($_POST[$questionvalue], NULL , $Thermal_Infrared)))
$counter_Thermal_Infrared++;
}}}

foreach($questions as $questionkey => $questionvalue) {
$Micro_Analog= file_get_contents("Micro_Analog.xml");
if( !empty( preg_match($_POST[$questionvalue], NULL , $Micro_Analog)))
$counter_Micro_Analog++;
}}}

$result=max($counter_Low_Light,$counter_HD_Video,$counter_Thermal_Infrared,$counter_Micro_Analog);
if ($result==$counter_Low_Light)
  echo " Low light camera is the best option for that condition" ;
if ($result==$counter_HD_Video)
  echo " HD Video camera is the best option for that condition" ;
if ($result==$counter_Thermal_Infrared)
  echo " Thermal Infrared camera is the best option for that condition" ;
if ($result==$counter_Micro_Analog)
  echo " Micro Analog camera is the best option for that condition" ;
?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html>
<head>
<title>Questionnaire</title>
</head>
<body>
<form name="SENSING MISSION MANAGEMENT PROGRAM(CAMERA)" method="post"
action="file:///C:/Users/hanzadeceylan/Desktop/cameras.php">
<p>SENSING MISSION MANAGEMENT PROGRAM(CAMERA)</p>
<table width="1073" border="0" bordercolor="#FFFFFF">
<tr>
<td width="357"><p>1.Features</p></td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list1" value="Wide-Angle Lens" />Wide-Angle Lens</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list1" value="Minimum Illumination" />Minimum Illumination</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list1" value="Thermal Imager" />Thermal Imager</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list1" value="Small and Light" />Small and Light</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list1" value="Full HD 1920x1080 Progressive" />Full HD 1920x1080 Progressive</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><p>2.Benefits</p></td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list2" value="Professional quality digital still photographs" />Professional quality digital still photographs</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list2" value="Full HD progressive digital video" />Full HD progressive digital video</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list2" value="Low light motion video" />Low light motion video</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list2" value="Thermal infrared motion video" />Thermal infrared motion video</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list2" value="Very low weight for longer flights" />Very low weight for longer flights</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><p>3.CCD / MOS /Sensor</p></td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list3" value="1/1.63 inch CCD" />1/1.63 inch CCD</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list3" value="1/6 inch 3CCD" />1/6 inch 3CCD</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list3" value="1/2 inch CCD" />1/2 inch CCD</td>
<td width="716"></td>
</tr>
<tr>
<td width="357"><input type="radio" name="list3" value="Vox Microbolometer" />Vox Microbolometer</td>
<td width="716"></td>
</tr>
</table>
</form>
</body>
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>4. Weight</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list4&quot; value=&quot;8.0oz / 227g&quot; /&gt;8.0oz / 227g&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list4&quot; value=&quot;11.6oz / 330g&quot; /&gt;11.6oz / 330g&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list4&quot; value=&quot;5.86oz / 166g&quot; /&gt;5.86oz / 166g&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list4&quot; value=&quot;5.4oz / 153g&quot; /&gt;5.4oz / 153g&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list4&quot; value=&quot;0.42oz / 12g&quot; /&gt;0.42oz / 12g&lt;br /&gt;</td>
</tr>
<tr>
<td>5. Max Pixels</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list5&quot; value=&quot;10.1 Megapixels&quot; /&gt;10.1 Megapixels&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list5&quot; value=&quot;2.1 Megapixel Digital Camera&quot; /&gt;2.1 Megapixel Digital Camera&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list5&quot; value=&quot;768x494 Effective Pixels&quot; /&gt;768x494 Effective Pixels&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list5&quot; value=&quot;324x256 Effective Pixels&quot; /&gt;324x256 Effective Pixels&lt;br /&gt;</td>
</tr>
<tr>
<td>6. Color</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list6&quot; value=&quot;Full Color&quot; /&gt;Full Color&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list6&quot; value=&quot;Monochrome&quot; /&gt;Monochrome&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list6&quot; value=&quot;False Color Option&quot; /&gt;False Color Option&lt;br /&gt;</td>
</tr>
<tr>
<td>7. Onboard Digital Video Recording Mode</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list7&quot; value=&quot;QuickTime Motion Photo JPEG&quot; /&gt;QuickTime Motion Photo JPEG&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list7&quot; value=&quot;AVCHD&quot; /&gt;AVCHD&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list7&quot; value=&quot;None&quot; /&gt;None&lt;br /&gt;</td>
</tr>
<tr>
<td>8. TV Out Playback Mode</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list8&quot; value=&quot;NTSC&quot; /&gt;NTSC&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list8&quot; value=&quot;PAL&quot; /&gt;PAL&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list8&quot; value=&quot;Selectable&quot; /&gt;Selectable&lt;br /&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list8&quot; value=&quot;Non selectable&quot; /&gt;Non selectable&lt;br /&gt;</td>
</tr>
<tr>
<td>9. Onboard Video / Image Storage</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;input type=&quot;radio&quot; name=&quot;list9&quot; value=&quot;SD / SDHC / MMC Memory Card&quot; /&gt;SD / SDHC / MMC Memory&lt;br /&gt;</td>
</tr>
</tbody>
</table>
Card<br />
<input type="radio" name="list9" value="SD / SDHC Memory Card" />SD / SDHC Memory Card<br />
<input type="radio" name="list9" value="None (wireless video transmit only)" />None (wireless video transmit only)<br />
</p> <p><br />
10.Zoom<br />
<input type="radio" name="list10" value="2.5x Optical Zoom" />2.5x Optical Zoom<br />
<input type="radio" name="list10" value="10x Optical Zoom" />10x Optical Zoom<br />
<input type="radio" name="list10" value="3x Optical Zoom" />3x Optical Zoom<br />
<input type="radio" name="list10" value="2x Digital Zoom" />2x Digital Zoom<br />
<input type="radio" name="list10" value="None" />None<br />
<br /></p><p></td><td width="395"><p>
11.Lens<br />
<input type="radio" name="list11" value="24 - 60mm Wide-Angle Lens" />24 - 60mm Wide-Angle Lens<br />
<input type="radio" name="list11" value="35mm Equivalent Leica Lens" />35mm Equivalent Leica Lens<br />
<input type="radio" name="list11" value="4mm" />4mm<br />
<input type="radio" name="list11" value="None" />None<br />
</p>
</td><td width="395"><p>
12.Focal Length<br />
<input type="radio" name="list12" value="5.1 to 12.8mm" />5.1 to 12.8mm<br />
<input type="radio" name="list12" value="3.0 to 30mm" />3.0 to 30mm<br />
<input type="radio" name="list12" value="4 to 12mm" />4 to 12mm<br />
<input type="radio" name="list12" value="21.5mm" />21.5mm<br />
</p>
</td><td width="395"><p>
13.Onboard Image Stabilization<br />
<input type="radio" name="list13" value="Yes" />Yes<br />
<input type="radio" name="list13" value="No" />No<br />
<p>&nbsp;&nbsp;</p>
</p><p>
14.ISO / Minimum Illumination <br />
<input type="radio" name="list14" value="80 to 6400 ISO" />80 to 6400 ISO<br />
<input type="radio" name="list14" value="5 Lux" />5 Lux<br />
</p>
<input type="radio" name="list14" value="0.0001 Lux" />
0.0001 Lux<br />

<input type="radio" name="list14" value=">50mK at f/1.0 with FLIR proprietary noise reduction" />
50mK at f/1.0 with FLIR proprietary noise reduction<br />

<input type="radio" name="list14" value="0.19 Lux" />
0.19 Lux<br />

<p>&nbsp;</p>

15. Type of data<br />

<input type="radio" name="list15" value="Video" />
Video<br />

<input type="radio" name="list15" value="Photograph" />
Photograph<br />

<br /></td><tr><td>
<label></label>
<input type="submit" name="Submit" value="Submit" />
<br/>
</p>
</form>
</body>
</html>