**2. XML Report**

|  |  |  |
| --- | --- | --- |
| **Changes from UML to XML** |  | In changing from a UML Design to an XML implementation, we didn't have to make many changes. One of the changes we made was adding the ‘Services’ class to the UML class diagram. We did this as we felt a services XML implementation would tie in better with our XML implementation of the Vehicle Maintenance class, and allowed us to write more useful and practical queries than we would have if we chose another existing UML class. |

|  |  |  |
| --- | --- | --- |
| **Team Roles** |  | **Brandon:** Designed the XML file ‘mechanics\_info.XML’ and the corresponding ‘mechanics\_info.dtd’. Designed 3 Xqueries, which can be found below under the user-defined functions *get\_mechanics\_payroll\_info(), get\_free\_mechanics()* and *get\_service\_history\_for($mechanicname as xs:string)*. Attended regular team meetings and contributed as a part of the team to all aspects of the development of the XML implementation for the system. Wrote the ‘*Strengths and Weaknesses’* section of the report found below.  **Cian:** Wrote the ‘XQueries’ section of the report, copying out the teams queries and briefly describing them. Worked with the team during group meetings to complete our set tasks. Designed the XML file ‘repairs.xml’ which stemmed from Michael’s vehicle maintenance file. Designed two XQueries, (query 4 and 5 in the Xquery section).  **Michael:** Designed the XML files *parts\_inventory.xml*, *vehicle\_maintenance.xml* and *services.xml*. In addition I designed 3 xquery queries, query 7 which returns the parts needed that are in stock for a vehicle service as well as query 8 which returns a list of the mechanics who’ve serviced a particular vehicle. Organised each team meeting for the xml design and implementation each week.  **Patrick William:** Designed the XML file Admin\_Info.xml, attended all group meetings to work on our given tasks and talk with the group. I also designed the XQuery 9 which checks if an admin has the user privilege of accessing a particular mechanic’s information by passing in the Admin’s user ID and the mechanic’s ID as parameters into a local function. I wrote the changes from UML design to XML implementation part for this report.  **Arthur:** Attended team meetings and contributed towards the discussion and design of our XML implementation. Created the Vehicle\_Management.xml file along with the Vehicle\_Management.dtd file. I also designed two XQueries, query 6, which shows all vehicles that need to be serviced and query 10, which checks when a service is due and for which vehicle. Also put the XML and DTD files into the report. |

|  |  |  |
| --- | --- | --- |
| **Strengths and Weaknesses** |  | **Strengths:** There are several strengths that can be associated with the design of both our XML model and the Xqueries designed upon it. Modelling the system in XML allows for a strict, defined layout of the system and all the parts within. It allows any external user (both with knowledge of XML and without) to look at the system and immediately grasp an understanding of what it entails and how it works. Using XML to model the system also allows for a graphical representation of the model to be generated via software programs such as BaseX. This allows for a much easier overview of the system as a whole and would be extremely beneficial for cases such as explaining how the system works to those of a non-technical background.  Using strict and well defined XML DTD’s ensures that the system maintains a rigid structure and that everything within the system has a specific place at which it can be found. The use of XML is also extremely beneficial when it comes to a user attempting to act on the system e.g in our case, a mechanic trying to register a service and its details. To perform these functions we implemented several Xqueries which act on the system and can be found in further detail below. Xqueries are very useful once the user has knowledge of how the XML design is structured. It allows the larger system to be accessed quickly and easily. It allows for user-defined functions to be created which allow actors on the system to fulfill otherwise complicated tasks extremely easily. Some examples of user-defined functions acting on our system can be found below. Overall, we believe that when used correctly, XML and Xquery together are extremely useful and powerful methods of creating and managing information systems.  **Weaknesses:** To any user that may not have a technical background XML might first appear extremely daunting. In comparison to UML, at first glance it provides a much less comprehensible overview of the system itself. Without using UML models you are fairly limited when it comes to representing the flow of operations within the larger system. This can be done much easier by using UML Sequence Diagram’s / Activity Diagrams. To an outsider with no underlying knowledge of XML or other languages of similar syntax understanding the Xqueries which act on the system may also be extremely difficult. |

**XML Files**

***Admin\_Info.xml***

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

*<!DOCTYPE Admin\_Info SYSTEM "Admin\_Info.dtd">*

*<Admin\_Info>*

*<Admin user\_id="102">*

*<user\_name>*

*<first\_name>Jim</first\_name>*

*<surname>Ryan</surname>*

*</user\_name>*

*<user\_role>System security</user\_role>*

*<user\_privileges>*

*<mechanic\_ids>*

*<mechanic\_id>40391</mechanic\_id>*

*<mechanic\_id>40392</mechanic\_id>*

*<mechanic\_id>40393</mechanic\_id>*

*<mechanic\_id>40394</mechanic\_id>*

*<mechanic\_id>24829</mechanic\_id>*

*</mechanic\_ids>*

*<vehicle\_ids>*

*<vehicle\_id>2019</vehicle\_id>*

*<vehicle\_id>4930</vehicle\_id>*

*<vehicle\_id>13984</vehicle\_id>*

*<vehicle\_id>1246</vehicle\_id>*

*<vehicle\_id>4958</vehicle\_id>*

*</vehicle\_ids>*

*</user\_privileges>*

*</Admin>*

*<Admin user\_id="202">*

*<user\_name>*

*<first\_name>Timmy</first\_name>*

*<surname>Bob</surname>*

*</user\_name>*

*<user\_role>System maintenance</user\_role>*

*<user\_privileges>*

*<mechanic\_ids>*

*<mechanic\_id>40391</mechanic\_id>*

*<mechanic\_id>40392</mechanic\_id>*

*<mechanic\_id>40393</mechanic\_id>*

*<mechanic\_id>40394</mechanic\_id>*

*<mechanic\_id>24829</mechanic\_id>*

*</mechanic\_ids>*

*<vehicle\_ids>*

*<vehicle\_id>2019</vehicle\_id>*

*<vehicle\_id>4930</vehicle\_id>*

*<vehicle\_id>13984</vehicle\_id>*

*<vehicle\_id>1246</vehicle\_id>*

*<vehicle\_id>4958</vehicle\_id>*

*</vehicle\_ids>*

*</user\_privileges>*

*</Admin>*

*<Admin user\_id="105">*

*<user\_name>*

*<first\_name>Fergus</first\_name>*

*<surname>Moyles</surname>*

*</user\_name>*

*<user\_role>Trainee Admin</user\_role>*

*<user\_privileges>*

*<vehicle\_ids>*

*<vehicle\_id>2019</vehicle\_id>*

*<vehicle\_id>4930</vehicle\_id>*

*<vehicle\_id>13984</vehicle\_id>*

*<vehicle\_id>1246</vehicle\_id>*

*<vehicle\_id>4958</vehicle\_id>*

*</vehicle\_ids>*

*</user\_privileges>*

*</Admin>*

*</Admin\_Info>*

***Mechanics\_info.xml***

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

<!DOCTYPE Mechanics\_Info SYSTEM "mechanics\_info.dtd">

<Mechanics\_Info>

<Mechanic mechanic\_id="40391">

<name>

<first\_name>Marsha</first\_name>

<surname>Knight</surname>

</name>

<details>

<address>

<number\_or\_name>The Orchard</number\_or\_name>

<street>Pinehill Letterkenny</street>

<town>Letterkenny</town>

<postcode>B21 U819</postcode>

<county>Donegal</county>

</address>

<PPSN>8218290C</PPSN>

<phone\_number>0831821336</phone\_number>

<e-mail\_address>knightMarsh2@gmail.com</e-mail\_address>

<bank\_details>

<IBAN>28191CBOIJ129</IBAN>

<BIC>61BOI2910</BIC>

</bank\_details>

</details>

<role>Senior Mechanic</role>

<vehicles\_serviced>

<Vehicle VehicleID='4930' NeedsService="NO" isBeingServiced="NO">

<LastService>21/06/2017</LastService>

</Vehicle>

<Vehicle VehicleID="2019" NeedsService="YES" isBeingServiced="NO">

<LastService>14/03/2016</LastService>

</Vehicle>

<Vehicle VehicleID="13984" NeedsService="YES" isBeingServiced="YES">

<LastService>9/11/2017</LastService>

</Vehicle>

<Vehicle VehicleID="1246" NeedsService="YES" isBeingServiced="NO">

<LastService>31/9/2017</LastService>

</Vehicle>

<Vehicle VehicleID="4958" NeedsService="NO" isBeingServiced="NO">

<LastService>23/8/2015</LastService>

</Vehicle>

</vehicles\_serviced>

<is\_servicing>false</is\_servicing>

</Mechanic>

<Mechanic mechanic\_id="40392">

<name>

<first\_name>Mark</first\_name>

<surname>O'Sullivan</surname>

</name>

<details>

<address>

<number\_or\_name>21</number\_or\_name>

<street>Seabury Crescent</street>

<town>Malahide</town>

<postcode>K28 C910</postcode>

<county>Dublin</county>

</address>

<PPSN>48183990D</PPSN>

<phone\_number>0851309736</phone\_number>

<e-mail\_address>mark.osul@yahoo.com</e-mail\_address>

<bank\_details>

<IBAN>73491CAIBK129</IBAN>

<BIC>10AIB2910</BIC>

</bank\_details>

</details>

<role>Junior Mechanic</role>

<vehicles\_serviced>

<Vehicle VehicleID="1246" NeedsService="NO" isBeingServiced="NO">

<LastService>31/9/2017</LastService>

</Vehicle>

</vehicles\_serviced>

<is\_servicing>false</is\_servicing>

</Mechanic>

<Mechanic mechanic\_id="40393">

<name>

<first\_name>Tim</first\_name>

<surname>Stanley</surname>

</name>

<details>

<address>

<number\_or\_name>95</number\_or\_name>

<street>Drimnagh Woods</street>

<town>Castlenock</town>

<postcode>W72 G731</postcode>

<county>Dublin</county>

</address>

<PPSN>38823938P</PPSN>

<phone\_number>0862279106</phone\_number>

<e-mail\_address>timmystan@live.co.uk</e-mail\_address>

<bank\_details>

<IBAN>10036CABOI190</IBAN>

<BIC>93BOI2170</BIC>

</bank\_details>

</details>

<role>Senior Mechanic</role>

<vehicles\_serviced>

<Vehicle VehicleID='4930' NeedsService="NO" isBeingServiced="NO">

<LastService>21/06/2017</LastService>

</Vehicle>

<Vehicle VehicleID="2019" NeedsService="YES" isBeingServiced="NO">

<LastService>14/03/2016</LastService>

</Vehicle>

<Vehicle VehicleID="13984" NeedsService="YES" isBeingServiced="YES">

<LastService>9/11/2017</LastService>

</Vehicle>

<Vehicle VehicleID="1246" NeedsService="YES" isBeingServiced="NO">

<LastService>31/9/2017</LastService>

</Vehicle>

<Vehicle VehicleID="4958" NeedsService="NO" isBeingServiced="NO">

<LastService>23/8/2015</LastService>

</Vehicle>

</vehicles\_serviced>

<is\_servicing>true</is\_servicing>

</Mechanic>

<Mechanic mechanic\_id="40394">

<name>

<first\_name>Bob</first\_name>

<surname>Curtis</surname>

</name>

<details>

<address>

<number\_or\_name>92</number\_or\_name>

<street>Streamstown Lane</street>

<town>Swords</town>

<postcode>H37 X198</postcode>

<county>Dublin</county>

</address>

<PPSN>72593702K</PPSN>

<phone\_number>0837703126</phone\_number>

<e-mail\_address>bobbycurtis420@gmail.com</e-mail\_address>

<bank\_details>

<IBAN>N/A</IBAN>

<BIC>N/A</BIC>

</bank\_details>

</details>

<role>Intern</role>

<vehicles\_serviced>

</vehicles\_serviced>

<is\_servicing>false</is\_servicing>

</Mechanic>

<Mechanic mechanic\_id="40395">

<name>

<first\_name>Willie</first\_name>

<surname>Walsh</surname>

</name>

<details>

<address>

<number\_or\_name>47</number\_or\_name>

<street>Raglan Road</street>

<town>Ballybough</town>

<postcode>B42 U829</postcode>

<county>Dublin</county>

</address>

<PPSN>14228602Z</PPSN>

<phone\_number>08912304126</phone\_number>

<e-mail\_address>williewalsher@yahoo.com</e-mail\_address>

<bank\_details>

<IBAN>26937CAIBK365</IBAN>

<BIC>64AIB4137</BIC>

</bank\_details>

</details>

<role>Apprentice Mechanic</role>

<vehicles\_serviced>

<Vehicle VehicleID="1246" NeedsService="YES" isBeingServiced="NO">

<LastService>31/9/2017</LastService>

</Vehicle>

</vehicles\_serviced>

<is\_servicing>true</is\_servicing>

</Mechanic>

</Mechanics\_Info>

***Parts\_inventory.xml***

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

<!DOCTYPE partsInventory SYSTEM "parts\_inventory.dtd">

<partsInventory>

<part id="6582092" category="brakes" stock="12">

<make>ford</make>

<model year="1995-2000">transit</model>

<name>Master-Sport Brake Discs (pair)</name>

<partDescription>To fit ford transit From Jan 1995 to Nov 2000 1.8[125hp 1781cc]</partDescription>

<fittingPosition>front axle</fittingPosition>

<RRP currency="euro">58.16</RRP>

</part>

<part id="1492666" category="filters" stock="5">

<make>mercedes</make>

<model year="1982-1992">sprinter</model>

<name>Mann Fuel Filter</name>

<partDescription>To fit Mercedes Sprinter from Sep 1982 to Mar 1992 316i[100hp 1596cc]</partDescription>

<RRP currency="euro">20.99</RRP>

<specification>

<height metric="mm">164</height>

<inlet metric="mm"></inlet>

<outerDiameter metric="mm">74</outerDiameter>

</specification>

</part>

<part id="1701812" category="filters" stock="5">

<make>hyundai</make>

<model year="2002-2009">i40</model>

<name>Wix Filtron Air Filter</name>

<partDescription>

To fit Hyundai i40 from Mar 2011 to Dec 2016 E220 CDI

(211.008)[170hp. 2148cc]

</partDescription>

<RRP currency="euro">22.99</RRP>

<specification>

<height metric="mm">434</height>

<outerDiameter metric="mm">120</outerDiameter>

<innerDiameter metric="mm">70</innerDiameter>

<shape>round</shape>

</specification>

</part>

<part id="1904414" category="clutch" stock="2">

<make>audi</make>

<model year="2003-2009">q7</model>

<name>Valeo Clutch Kit</name>

<partDescription>

To fit Audi Q7 from Oct 2015 to Feb 2017 2.0

GTI[200hp 1984cc.]

</partDescription>

<RRP currency="euro">377.99</RRP>

<specification>

<transmissionType>manual</transmissionType>

<engineCode>CDLG</engineCode>

<weight metric="kg">6.88</weight>

</specification>

</part>

<part id="49381405" category="tyres" stock="20">

<make>ford</make>

<model year="2003-2009">transit</model>

<name>Low profile Continental Tyre</name>

<partDescription>

To fit Ford Transit from Oct 2015 to Feb 2017 2.0

GTI[200hp 1984cc.]

</partDescription>

<RRP currency="euro">79.99</RRP>

</part>

<part id="4959383" category="gearbox" stock="1">

<make>mercedes</make>

<model year="2003-2009">sprinter</model>

<name>Mercedes Sprinter replacement gearbox</name>

<partDescription>

To fit Mercedes Sprinter from Oct 2015 to Feb 2017 2.0

GTI[200hp 1984cc.]

</partDescription>

<RRP currency="euro">459.99</RRP>

</part>

</partsInventory>

***repairs.xml***

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

<!DOCTYPE repairs SYSTEM "repairs.dtd">

<repairs>

<repair id="5689">

<vehicleID>2019</vehicleID>

<repairDescription>Flat tyres</repairDescription>

<repairType>Part replacement</repairType>

<partsNeeded>

<part id="49381405" quantity="4" />

</partsNeeded>

</repair>

<repair id="140596">

<vehicleID>4930</vehicleID>

<repairDescription>Door lock on passenger door broken</repairDescription>

<repairType>Part repair</repairType>

</repair>

<repair id="1458">

<vehicleID>13984</vehicleID>

<repairDescription>broken gear box</repairDescription>

<repairType>Part Replacment</repairType>

<partsNeeded>

<part id="4959383" quantity="1" />

</partsNeeded>

</repair>

</repairs>

***Services.xml***

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

<!DOCTYPE services SYSTEM "services.dtd">

<services>

<service id="40292">

<vehicleID>2019</vehicleID>

<mechanicID>40391</mechanicID>

<serviceDescription>

Regular minor service maintenance. New front and rear disk breaks also needed.

</serviceDescription>

<serviceType>Minor Service</serviceType>

<partsNeeded>

<part id="6582092" quantity="2" />

<part id="1492666" />

</partsNeeded>

</service>

<service id="402918">

<vehicleID>13984</vehicleID>

<mechanicID>40393</mechanicID>

<serviceDescription>

Regular major service at 150,000 km. Drivers complaining of vehicle pulling

to the left at higher speeds. Check as part of major service

</serviceDescription>

<serviceType>Major Service</serviceType>

</service>

<service id="30291">

<vehicleID>1246</vehicleID>

<mechanicID>24829</mechanicID>

<serviceDescription>

New air and fuel filters need to be fitted as well as major service

</serviceDescription>

<serviceType>Minor Service</serviceType>

<partsNeeded>

<part id="1492666" quantity="3" />

<part id="1701812" />

</partsNeeded>

</service>

</services>

***Vehicle\_maintenance.xml***

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

<!DOCTYPE vehicleMaintenance SYSTEM "vehicle\_maintenance.dtd">

<vehicleMaintenance>

<service id="40292">

<isBeingServiced>No</isBeingServiced>

<mechanics>

<mechanicID>40391</mechanicID>

</mechanics>

<priority>high</priority>

<dueDate>21-1-2018</dueDate>

</service>

<service id="402918">

<isBeingServiced>YES</isBeingServiced>

<mechanics>

<mechanicID>40393</mechanicID>

</mechanics>

<priority>low</priority>

<dueDate>12-2-2018</dueDate>

</service>

<service id="30291">

<isBeingServiced>NO</isBeingServiced>

<mechanics>

<mechanicID>24829</mechanicID>

</mechanics>

<priority>medium</priority>

<dueDate>12-12-2017</dueDate>

</service>

<repair id="5689">

<isBeingRepaired>NO</isBeingRepaired>

<priority>low</priority>

<dueDate>15-12-2017</dueDate>

</repair>

<repair id="140596">

<isBeingRepaired>YES</isBeingRepaired>

<mechanics>

<mechanicID>40392</mechanicID>

<mechanicID>40394</mechanicID>

</mechanics>

<priority>high</priority>

<dueDate>8-12-2017</dueDate>

</repair>

<repair id="1458">

<isBeingRepaired>NO</isBeingRepaired>

<priority>high</priority>

<dueDate>10-12-2017</dueDate>

</repair>

</vehicleMaintenance>

***Vehicle\_Management.xml***

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

<!DOCTYPE Vehicle\_Management SYSTEM "Vehicle\_Management.dtd">

<Vehicle\_Management>

<Vehicle VehicleID="2019" NeedsService="YES" isBeingServiced="YES">

<VehicleDetails>

<Type>Ambulance</Type>

<Make>Ford</Make>

<Model>Transit</Model>

<Year>2010</Year>

<RegNo>10-D-34056</RegNo>

<Mileage>342116</Mileage>

<Colour>Yellow</Colour>

</VehicleDetails>

</Vehicle>

<Vehicle VehicleID="4930" NeedsService="YES" isBeingServiced="NO">

<VehicleDetails>

<Type>Ambulance</Type>

<Make>Ford</Make>

<Model>Transit</Model>

<Year>2017</Year>

<RegNo>171-KE-133</RegNo>

<Mileage>60539</Mileage>

<Colour>Yellow</Colour>

</VehicleDetails>

</Vehicle>

<Vehicle VehicleID="13984" NeedsService="NO" isBeingServiced="NO">

<VehicleDetails>

<Type>Ambulance</Type>

<Make>Mercedes-Benz</Make>

<Model>Sprinter</Model>

<Year>2012</Year>

<RegNo>12-OY-12053</RegNo>

<Mileage>246433</Mileage>

<Colour>Yellow</Colour>

</VehicleDetails>

</Vehicle>

<Vehicle VehicleID="1246" NeedsService="NO" isBeingServiced="NO">

<VehicleDetails>

<Type>Garda Car</Type>

<Make>Hyundai</Make>

<Model>i40</Model>

<Year>2015</Year>

<RegNo>151-D-16</RegNo>

<Mileage>125977</Mileage>

<Colour>White</Colour>

</VehicleDetails>

</Vehicle>

<Vehicle VehicleID="4958" NeedsService="YES" isBeingServiced="YES">

<VehicleDetails>

<Type>Garda Car</Type>

<Make>Audi</Make>

<Model>Q7</Model>

<Year>2017</Year>

<RegNo>171-L-589</RegNo>

<Mileage>57932</Mileage>

<Colour>Black</Colour>

</VehicleDetails>

</Vehicle>

</Vehicle\_Management>

|  |  |
| --- | --- |
| **DTD Files** |  |

***Admin\_Info.dtd***

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

<!-- This document contains the XML Model for the info of all Admins within the larger Emergency Services Vehicle Maintenance System model -->

<!-- The + cardinality is used for Admin as there must be at least one Admin registered in the system or an infinite amount of Admins-->

<!ELEMENT Admin\_Info (Admin)+>

<!-- Each Admin has a required attribute 'user\_id', so if any changes are made to the system the ID of the Admin can be used as a reference. Each Admin can have multiple roles however must have at least one, hence the + cardinality is used. The cardinality \* is used for 'user\_privileges' as if no priveleges are specified, the Admin has root access to everything in the system. -->

<!ELEMENT Admin (user\_name, user\_role+, user\_privileges\*)>

<!ATTLIST Admin user\_id CDATA #REQUIRED>

<!ELEMENT user\_id (#PCDATA)>

<!ELEMENT user\_name (first\_name, surname)>

<!ELEMENT first\_name (#PCDATA)>

<!ELEMENT surname (#PCDATA)>

<!ELEMENT user\_role (#PCDATA)>

<!-- The user privileges 'mechanic\_ids' and 'vehicle\_ids' has the cardinality ? as not all Admins necessarily have access to both or any of them. 'mechanic\_id' and 'vehicle\_id' has the cardinality + as there must be at least one of each in the database. -->

<!ELEMENT user\_privileges (mechanic\_ids?, vehicle\_ids?)>

<!ELEMENT mechanic\_ids (mechanic\_id+)>

<!ELEMENT mechanic\_id (#PCDATA)>

<!ELEMENT vehicle\_ids (vehicle\_id+)>

<!ELEMENT vehicle\_id (#PCDATA)>

***Mechanics\_info.dtd***

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

<!-- This document contains the XML Model for the records/info of all mechanics within the larger Emergency Services Vehicle Maintenance System model -->

<!-- The \* cardinality is used for Mechanic as there can be no mechanics registered in the system or an infinite amount of mechanics-->

<!ELEMENT Mechanics\_Info (Mechanic)\*>

<!-- Each mechanic has an associated set of data attributed to the mechanic itself. Each mechanic has a required attribute 'mechanic\_id', this is so anyone operating on the system can effeciently and quickly access/edit a mechanics respective details. Each mechanic can have multiple roles however must have atleast one, hence the + cardinality is used. -->

<!ELEMENT Mechanic (name, details, role+, vehicles\_serviced, is\_servicing)>

<!ATTLIST Mechanic mechanic\_id CDATA #REQUIRED>

<!ELEMENT mechanic\_id (#PCDATA)>

<!ELEMENT name (first\_name,surname)>

<!ELEMENT first\_name (#PCDATA)>

<!ELEMENT surname (#PCDATA)>

<!ELEMENT details (address, PPSN, phone\_number, e-mail\_address, bank\_details)>

<!ELEMENT address (number\_or\_name, street, town, postcode, county)>

<!ELEMENT number\_or\_name (#PCDATA)>

<!ELEMENT street (#PCDATA)>

<!ELEMENT town (#PCDATA)>

<!ELEMENT postcode (#PCDATA)>

<!ELEMENT county (#PCDATA)>

<!ELEMENT PPSN (#PCDATA)>

<!ELEMENT phone\_number (#PCDATA)>

<!ELEMENT e-mail\_address (#PCDATA)>

<!ELEMENT bank\_details (IBAN, BIC)>

<!ELEMENT IBAN (#PCDATA)>

<!ELEMENT BIC (#PCDATA)>

<!ELEMENT role (#PCDATA)>

<!-- Each mechanic also has an associated record of all the vehicles he/she has previously serviced. This is to enable a sense of traceability to the system, so that in the event of an accident occuring as a direct result of a faulty service the full service history can be traced back. -->

<!ELEMENT vehicles\_serviced (Vehicle)\*>

<!-- Each vehicle has associated required attributes 'VehicleID', 'NeedsService', 'isBeingServiced' this is in place to allow effecient and quick access to a vehicles record by someone acting on the system. Each vehicle also has other pieces of data associated with it such as when it was last serviced, whether or not it needs a service and whether or not it is currently being serviced. -->

<!ELEMENT Vehicle (VehicleDetails, LastService)>

<!ATTLIST Vehicle VehicleID CDATA #REQUIRED>

<!ATTLIST Vehicle NeedsService CDATA #REQUIRED>

<!ATTLIST Vehicle isBeingServiced CDATA #REQUIRED>

<!ELEMENT VehicleID (#PCDATA)>

<!ELEMENT NeedsService (#PCDATA)>

<!ELEMENT isBeingServiced (#PCDATA)>

<!ELEMENT LastService (#PCDATA)>

<!ELEMENT is\_servicing (#PCDATA)>

***Parts\_inventory.dtd***

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

<!-- This document contains the design for a file containing parts inventory data. -->

<!-- The parts inventory contains a list of all parts in stock, information pertaining

the parts and stock figures. -->

<!ELEMENT partsInventory (part+)>

<!-- Each part with a unique part id has its own part entry. The id is entered as an attribute,

as is the category of the part and the no. of that part in stock. Make, model, name and part

description are all required elements for each part. ? is used for all others as they are

not required but can only occur a maximum of once. -->

<!ELEMENT part (make,model,name,partDescription,fittingPosition?,RRP?,specification?)>

<!ATTLIST part id CDATA #REQUIRED>

<!ATTLIST part category CDATA "uncategorized">

<!ATTLIST part stock CDATA #REQUIRED>

<!-- Make and model are required elements for each part and the model has a required attribute

to specify the model years for the part. -->

<!ELEMENT make (#PCDATA)>

<!ELEMENT model (#PCDATA)>

<!ATTLIST model year CDATA #REQUIRED>

<!-- The name and part description are also required for each part entry. -->

<!ELEMENT name (#PCDATA)>

<!ELEMENT partDescription (#PCDATA)>

<!-- Fitting position allows to specify which part of the vehicle the part is to be fitted to -->

<!ELEMENT fittingPosition (#PCDATA)>

<!-- RRP specifies the recommended retail price as reference for mechanics as to what a

reasonable price to pay and charge for a part is. The currency attribute can also be

specified and its default is euro. -->

<!ELEMENT RRP (#PCDATA)>

<!ATTLIST RRP currency CDATA "euro">

<!-- The specification element is used to provide important part specific information to a mechanic. -->

<!-- Some elements such as height and diameters have a metric attribute to specify the unit of measurement -->

<!-- All elements have a cardinality of ? as they are all optional elements but may only occur a maximum of once-->

<!ELEMENT specification (height?,inlet?,outerDiameter?,innerDiameter?,shape?,transmissionType?,engineCode?,weight?)>

<!ELEMENT height (#PCDATA)>

<!ATTLIST height metric CDATA "mm">

<!ELEMENT inlet (#PCDATA)>

<!ATTLIST inlet metric CDATA "mm">

<!ELEMENT outerDiameter (#PCDATA)>

<!ATTLIST outerDiameter metric CDATA "mm">

<!ELEMENT innerDiameter (#PCDATA)>

<!ATTLIST innerDiameter metric CDATA "mm">

<!ELEMENT shape (#PCDATA)>

<!ELEMENT transmissionType (#PCDATA)>

<!ELEMENT engineCode (#PCDATA)>

<!ELEMENT weight (#PCDATA)>

<!ATTLIST weight metric CDATA "kg">

***repairs.dtd***

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

<!-- This document contains the design for a file containing vehicle repairs data. -->

<!ELEMENT repairs (repair+)>

<!-- Not all repairs will have parts needed so it is marked with ? cardinality. -->

<!ELEMENT repair (vehicleID, repairDescription, repairType, partsNeeded?)>

<!ATTLIST repair id CDATA #REQUIRED>

<!ELEMENT vehicleID (#PCDATA)>

<!ELEMENT repairDescription (#PCDATA)>

<!ELEMENT repairType (#PCDATA)>

<!-- Each parts needed contains parts elements with id attributes and quantity attributes.

The id attribute is required and the default quantity is 1 -->

<!ELEMENT partsNeeded (part+)>

<!ELEMENT part EMPTY>

<!ATTLIST part id CDATA #REQUIRED>

<!ATTLIST part quantity CDATA "1">

***services.dtd***

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

<!-- This document contains the design for a file containing vehicle maintenance data. -->

<!-- The parts inventory contains a list of all services to be completed, information pertaining

the service and mechanic assigned to the service. The + cardinality is used for service as

there must be one but there can be as many as needed -->

<!ELEMENT services (service+)>

<!-- Each service contains important information for the mechanic. The vehicle ID is required,

however the mechanic ID is not (marked with ? cardinality). This is because the service could

be added without a mechanic initially assigned to it. The same goes for parts needed. There

may not always be parts needed to complete the service. -->

<!-- The service ID is an attribute to the service element -->

<!ELEMENT service (vehicleID, mechanicID?, serviceDescription, serviceType, partsNeeded?)>

<!ATTLIST service id CDATA #REQUIRED>

<!ELEMENT vehicleID (#PCDATA)>

<!ELEMENT mechanicID (#PCDATA)>

<!ELEMENT serviceDescription (#PCDATA)>

<!ELEMENT serviceType (#PCDATA)>

<!-- Each parts needed contains parts elements with id attributes and quantity attributes.

The id attribute is required and the default quantity is 1 -->

<!ELEMENT partsNeeded (part+)>

<!ELEMENT part EMPTY>

<!ATTLIST part id CDATA #REQUIRED>

<!ATTLIST part quantity CDATA "1">

***vehicle\_maintenance.dtd***

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

<!-- This document contains the design for a file containing vehicle maintenance data. -->

<!-- The parts inventory contains a list of all services to be completed, information pertaining

the service and mechanic assigned to the service. The + cardinality is used for service as

there must be one but there can be as many as needed -->

<!ELEMENT vehicleMaintenance (service\*, repair\*)>

<!-- Each service contains important information for the mechanic. The vehicle ID is required,

however the mechanic ID is not (marked with ? cardinality). This is because the service could

be added without a mechanic initially assigned to it. The same goes for parts needed. There

may not always be parts needed to complete the service. -->

<!-- The service ID is an attribute to the service element -->

<!ELEMENT service (isBeingServiced, mechanics?, priority?, dueDate?)>

<!ATTLIST service id CDATA #REQUIRED>

<!-- Each repair contains important information for the mechanic. The vehicle ID is required,

however the mechanic ID is not (marked with ? cardinality). This is because the repair could

be added without a mechanic initially assigned to it. The same goes for parts needed. There

may not always be parts needed to complete the repair. -->

<!-- The repair ID is an attribute to the service element -->

<!ELEMENT repair (isBeingRepaired, mechanics?, priority?, dueDate?)>

<!ATTLIST repair id CDATA #REQUIRED>

<!ELEMENT isBeingRepaired (#PCDATA)>

<!ELEMENT isBeingServiced (#PCDATA)>

<!ELEMENT mechanics (mechanicID+)>

<!ELEMENT priority (#PCDATA)>

<!ELEMENT dueDate (#PCDATA)>

<!ELEMENT mechanicID (#PCDATA)>

***Vehicle\_Management.dtd***

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

<!-- This document contains the design for the vehicle management data.

The vehicle management class contains the vehicles that are in the fleet.

The \* is used for the Vehcile as there must be at least one vehcile in the system

but cannot be an infinite amount. -->

<!ELEMENT Vehicle\_Management (Vehicle)\*>

<!-- Each vehcile requires to have the attributes VehicleID, NeedsService and isBeingServiced.

These attributes are required so the mechanics can easily access the vehicles to see whether service is needed and if they have already been serviced.

Each vehicle also has a list of its details which include the type of vehicle, the make, model, year, registration number, mileage and colour. These details are also accessed by the mechanics -->

<!ATTLIST Vehicle VehicleID CDATA #REQUIRED>

<!ATTLIST Vehicle NeedsService CDATA #REQUIRED>

<!ATTLIST Vehicle isBeingServiced CDATA #REQUIRED>

<!ELEMENT Vehicle (VehicleDetails)>

<!ELEMENT VehicleDetails (Type, Make, Model, Year, RegNo, Mileage, Colour)>

<!ELEMENT Type (#PCDATA)>

<!ELEMENT Make (#PCDATA)>

<!ELEMENT Model (#PCDATA)>

<!ELEMENT Year (#PCDATA)>

<!ELEMENT RegNo (#PCDATA)>

<!ELEMENT Mileage (#PCDATA)>

<!ELEMENT Colour (#PCDATA)>

|  |  |
| --- | --- |
| **XQueries** |  |

***Query 1*** : **Return payroll into for all mechanics**

*This query uses a user defined function to search for the mechanics and from there it can return the payroll info.*

declare function local:get\_mechanics\_payroll\_info()

{

let $c:= doc("mechanics\_info.XML")

for $m in $c/Mechanics\_Info/Mechanic

let $firstname := $m/name/first\_name/text()

let $surname := $m/name/surname/text()

let $fullname := concat($firstname," ",$surname)

return

<payroll\_info>

<mechanic id ="{ $m/@mechanic\_id }" >

<name> {$fullname} </name>

{$m/details/bank\_details}

</mechanic>

</payroll\_info>

};

<Mechanics\_Payroll\_Info>

{local:get\_mechanics\_payroll\_info()}

</Mechanics\_Payroll\_Info>

***Query 2*** : **Get all free mechanics**

*This query searches for mechanics who are currently available to do work by checking if the /is\_servicing data is false and will return all mechanics who are currently free.*

declare function local:get\_free\_mechanics()

{

let $c:= doc("mechanics\_info.XML")

for $m in $c/Mechanics\_Info/Mechanic

where $m/is\_servicing/text() = "false"

let $firstname := $m/name/first\_name/text()

let $surname := $m/name/surname/text()

let $fullname := concat($firstname," ",$surname)

return

<mechanic id ="{ $m/@mechanic\_id }" >

<name> {$fullname} </name>

</mechanic>

};

<Free\_Mechanics>

{local:get\_free\_mechanics()}

</Free\_Mechanics>

***Query 3*** : **Get mechanics service history**

*This query searches for a certain mechanic by name and will return all of their service history by using a user defined function.*

declare function local:get\_service\_history\_for($mechanicname as xs:string)

{

let $c:= doc("secondyear/mt/infoman\_cs2041/assignments/assignment\_2\_XML/mechanics\_info.XML")

for $m in $c/Mechanics\_Info/Mechanic

let $fullname := concat($m/name/first\_name/text()," ",$m/name/surname/text())

where $fullname = $mechanicname

return

<mechanic id ="{ $m/@mechanic\_id }" >

<name> {$fullname} </name>

{$m/vehicles\_serviced}

</mechanic>

};

<Service\_History>

{local:get\_service\_history\_for("Mark O'Sullivan")}

</Service\_History>

***Query 4*** : **Find high priority vehicles in need of service**

*This query uses two xml databases to figure out what vehicles are high priority and then returns the corresponding information for the service required.*

declare function local:get\_high\_priority() {

for $s in doc("services.xml")/services/service,

$v in doc("vehicle\_maintenance.xml")/vehicleMaintenance/service

where $v/priority = "high"

where $v/string(@id) = $s/string(@id)

return $s

};

<vehicles>

{local:get\_high\_priority()}

</vehicles>

***Query 5*** : **Check stock for a part in inventory**

*This query searches the category attribute for a certain part and will return details about that part and how many there are in stock*

declare function local:get\_part\_category($part\_category as xs:string) {

for $s in doc("parts\_inventory.xml")/partsInventory/part

where contains($s/@category, $part\_category)

return

<part>

{$s/@category}

{$s/name}

{$s/partDescription}

<stock>{string($s/@stock)}</stock>

{$s/RRP}

<partID>{$s/@id}</partID>

</part>

};

<parts>

{local:get\_part\_category("clutch")}

</parts>

***Query 6*** : **What vehicles need servicing**

*This query uses three xml databases to return the ID and details about the vehicle if it is in need of service.*

for $j in /vehicleMaintenance/service,

$i in /services/service,

$x in /Vehicle\_Management/Vehicle

where $j/string(@id) = $i/string(@id)

where $i/string(vehicleID) = $x/string(@VehicleID)

return

<Needs\_Service>

{$i/vehicleID}

{$x/VehicleDetails}

</Needs\_Service>

***Query 7*** : **Check for parts in stock for a particular service**

*This query first checks if a vehicle is being serviced and then it checks what parts are needed and then if they are in stock.*

declare function local:partsInStock($service\_id as xs:string) {

for $servicesInMaintenance in doc("vehicle\_maintenance.xml")/vehicleMaintenance/service[@id = $service\_id],

$services in doc("services.xml")/services/service,

$partsInStock in doc("parts\_inventory.xml")/partsInventory/part

where $servicesInMaintenance/@id = $services/@id

where $services/partsNeeded/part/@id = $partsInStock/@id

return $partsInStock

};

<parts>

{local:partsInStock("40292")}

</parts>

***Query 8*** : **For Vehicle x print mechanics who’ve serviced it**

*This query uses the vehicle ID to find out what mechanics have serviced it and it will return their mechanic ID.*

declare function local:mechanicsServiced($vehicle\_id as xs:string) {

for $mechanic in doc("Mechanics\_Info.xml")/Mechanics\_Info/Mechanic

where $mechanic/vehicles\_serviced/Vehicle/string(@VehicleID) = $vehicle\_id

return

<mechanic>

{$mechanic/name}

<mechanicID>{$mechanic/string(@mechanic\_id)}</mechanicID>

</mechanic>

};

<mechanics>

{local:mechanicsServiced("2019")}

</mechanics>

***Query 9*** : **Check if admin has privileges for mechanic**

*This query uses an if statement to find out if a certain mechanic has admin privileges by comparing their ID’s in both the mechanic database and the admin database.*

declare function local:checkPrivileges($adminid as xs:string, $mechid as xs:string) {

let $admin := doc("Admin\_Info.xml")/Admin\_Info/Admin[@user\_id = $adminid]

for $mechanicIds in $admin/user\_privileges/mechanic\_ids/mechanic\_id

where contains(string($mechanicIds), $mechid)

return if ($mechanicIds)

then "true"

else "false"

};

local:checkPrivileges("102", "40391")

***Query 10*** : **Checks when a service is due and for which vehicle**

This query uses the vehicle\_maintence.xml and services.xml files to find the vehicles that need to be serviced through their service ID’s and return when that vehicle is due a service.

for $j in /vehicleMaintenance/service,

$i in /services/service

where $j/string(@id) = $i/string(@id)

let $c := $j/string(dueDate)

let $v := $i/string(vehicleID)

return

<Vehicle id ="{$v}">

ServiceDue: {$c}.

</Vehicle>