Software Requirements Specification (SRS)
Paint Defect Analysis Tool (PDAT)
Authors: Team 12 -- Kyutae Park, Christopher Cho, Kevin Ahn, Jon Hayward, Vikram Thakur
Customer: General Motors
Instructor: James Daly

1 Introduction
In this document, the Paint Defect Analysis Tool, or PDAT, shall be covered in complete
detail. The reasons for its production, the parts one would need to use it, the requirements it must meet,
and detailed specifications will all be discussed by this document. Also, the features of a prototype will
also be covered. This document will describe the entire PDAT system and its major functions as well as
the development process behind it. It will introduce potential constraints and discuss the dependencies
of the system in regards to the defect analysis occurring in an automotive paint line. Technical models are
also provided and a prototype is discussed in order to create guidelines as to how the system will be built.

1.1 Purpose
The purpose of this document is to catalog the user requirements, tools required, and motivations
for the paint defect analysis tool being developed by the team. This document is meant to help facilitate
and reach an agreement between the end user and developer, the system architect and the implementer, as
well as the programmer using the module and programmer implementing the module.

1.2 Scope
The software being produced is a system that is meant to automate the aggregation and generation
of an automotive paint defect analysis tool. This does not involve replacing an employee on the plant
floor checkpoint areas, however it does eliminate the use of a paper diagrams to record paint defects.
PDAT is the application tool being built and the domain of its development is only program being
developed. The new implemented system requires an employee to use a tablet to input any defects that
were previously recorded by hand. These reports can be automatically collated to create quality analysis
reviews and reports. By creating a digital system, we enable an increase in speed, reliability, security, and
portability. The objective of the system is to improve overall efficiency in the process of creating analysis
reports. The software will not be automatically gathering input data via sensors or other aids, it will still
require an employee to monitor vehicles. The software will also generate reports by request of the user
and be able to apply each certain filters to organize each report. Ultimately, the elimination of a reliance
on paper and manual analysis will result in a streamlined tool and service.

1.3 Definitions, acronyms, and abbreviations

<table>
<thead>
<tr>
<th>Required Terms, Definitions, or Abbreviations</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDAT</td>
<td>Application name, acronym for Paint Defect Analysis Tool</td>
</tr>
<tr>
<td>User Interface (UI)</td>
<td>The means by which the user and a computer system interact, in particular the use of input devices and software</td>
</tr>
<tr>
<td>Horizontal vs Vertical (defect location)</td>
<td>The sides of the vehicle are vertical, while the roof and hood are horizontal</td>
</tr>
<tr>
<td>Employee Verification</td>
<td>A process to authenticate users. This will determine how much access the user is given based on their status</td>
</tr>
</tbody>
</table>
E-Coat (Electrocoating)  |  Process used for coating a wide range of consumer goods including hardware, jewelry, eyeglass frames, giftware and many other items
---|---
Car name and model abbreviations  |  Specify abbreviations used to describe the name of the car as well as the model
PDU  |  Paint Defects Per Unit
Paint defect location abbreviations/descriptions  |  Specify abbreviations used to describe the location of the paint defect
Polish Deck, Prime review checkpoints  |  Different checkpoints to analyze the quantity and severity of the paint defects
Audit  |  The electronic equivalent of the paper diagrams that a client analyst will fill out when inspecting vehicles

Examples that contain other terms can be found at:

### 1.4 Organization

The rest of the SRS document will contain an overall description of the Paint Defect Analysis Tool, models that describe the software in a more technical depth, a functionality specification for the prototype and the references used throughout the development of the SRS document.

### 2 Overall Description

In this section, the aspects of PDAT not directly tied to how it performs will be discussed. The context of the product, PDAT’s functions, the users of PDAT, the team’s assumptions, constraints for PDAT, and requirements that will not be fulfilled by PDAT are discussed at length in this portion of the document.

#### 2.1 Product Perspective

The PDAT will provide a way to improve the data entry method of paint defects on an automotive line. This will contribute to the overall paint assembly line system and will utilize other systems throughout the manufacturing plant. The solution that the analysis tool will provide must have specific characteristics. A friendly user interface is essential to increasing efficiency of data input. This also allows for a wider range of skilled employees to interact with the software.

The user interface will be directed towards the defect recorder. Thus the application will have an authentication page, a page selecting the type of vehicle, and a page that will allow the user to submit defects on a specific vehicle. On each review line, a client analyst will log into the application, navigate
the menu and select if they would like to view a report or log data entries. If the analyst decides to log data entry then they will proceed to select the current vehicle they are analyzing and begin to mark defects. Each defect type will be chosen via a toolbar and it will be recorded upon press to the vehicle diagram on the screen. This data will automatically saved, so the employee may feel free to move between different vehicle diagrams as they come down the line. After the data is saved, it will contribute to the different reports that are then generated automatically. These reports will also be able to be accessed whenever needed via the opening application menu.

By automating the paint defect analysis tool, the program will eliminate unexpected difficulties such as unreadable handwriting or ambiguous defect marks. The tool must also meet specific performance requirements for both the web application and the report generator to create a seamless experience. The collation of the vehicle diagrams must be automated. The quality analysis reports must also be generated autonomously. These reports must be able to be created with any given timeframe. To ensure the integrity of the reports, only users with correct security credentials will have access.

The PDAT tool will require synergy between multiple systems. To eliminate the use of paper, a web application will be used via tablet system. This ensures a more secure method of reporting and storing data. The analysis tool requires a connection to the plant’s network to interact with outside systems and improve communication. To further secure the reports, an employee credential system will need to be accessed in order to create reports and analyses. The report’s data will be stored in a database system. A diagram of the system interactions with each other is shown below in figure 2-1.

Because the system relies on other systems and sensitive manufacturing data, there are constraints that exist. Secure access to data and the systems must be ensured in order reserve integrity. Another form of constraints that exists is team constraints which include organization of time relative to other classes, meetings, and work load throughout the development process. Meetings and equal workload should be organized weekly and effectively for entire project growth.
2.2 Product Functions
The major functions that PDAT will perform include data mining what the user inputs and authenticating the user for access. Furthermore, data mining will take all the inputs, entered by the user using a tablet, and will output the data to be analyzed. Once the new data have been created, the software will generate charts to visualize how these values have been changing weekly. As per authentication, if the user does not have access to the software, the software will prevent the user to enter any data. This data will include location, type, and severity of defects. As per authentication, if the user does not have access to PDAT, it will prevent the user to enter any data. The customer specified that PDAT would be compatible in all three of the GM plants, so the output of the data must account for different systems being used to generate the desired reports in all three plants. Also, the functionalities specified by the user include a friendly user interface and collation of the vehicle diagrams.

2.3 User Characteristics
There are two expected users of PDAT that often overlap. The first user is the inspector that will be located on the line at a checkpoint. This user will look at vehicles and use the system to report defects in the paint. This user will, if this person is not also the other user, not expected to have significant
education beyond high school. This person will be skilled with their eyes, but is not expected to be proficient with computers if this is the only job the worker has.

The other expected user is the analyst. This user will take the data gathered by the inspector, which will usually be the analyst, and then generate reports. The analyst will then comb through the reports to determine trends and find solutions for what is causing the defects. This person will most likely have a college degree and will have extensive experience with computers.

2.4 Constraints

Due to PDAT’s reliance on external systems, constraints exist beyond the control of the software. PDAT may be constrained due to safety concerns as well. A user creates and fills out an audit through an app on a tablet. The tablet must be able to run the app to provide a smooth and stable user experience. PDAT does not store data on the tablet directly but in a database elsewhere. To access this database, the tablet must have an internet connection to submit audits, request to generate reports, and view completed reports. For security concerns, only users with verified credentials have access to creating audits and dealing with reports. This authentication is verified by General Motor’s pre-existing employee verification system.

2.5 Assumptions and Dependencies

The hardware that PDAT is run on will primarily be a tablet that the client analyst will use when checking cars for paint defects. The system will also depend on the database that holds all stored reports. PDAT will also expect some sort of authentication system provided by each plant. PDAT will have to interface with file creation and a printer. User interactions with PDAT should be restrained to analysts making inputs of defects and then generating reports based on these defects.

2.6 Approportioning of Requirements

The current system is meant to interface a system in order for data entry by employees for defects and generate reports. In addition, PDAT will still require an employee to monitor vehicles to identify and enter the three data required for analysis: location, type, and severity. The employee must be verified through an existing authentication system in order to have the correct permissions to view data and manipulate data.

3 Specific Requirements

1. Receive Paint Defect Data
   1.1. Data must include location of defect
   1.2. Data must include severity of defect
   1.3. Data must include the model of the car the defect occurred on
   1.4. Data must include the line number and plant the car was in
   1.5. Data must be able to be removed before it is stored

2. Generate Reports
   2.1. Report durations supported: daily, weekly, monthly
   2.2. Report types supported: Q.A, Audit, Summary Report

3. Store Data
   3.1. Data must be stored in a database
   3.1.1. The database must be secure
   3.1.2. The database must keep records indefinitely
   3.2. Data must be retrievable for the generation of reports

4. Log In
   4.1. Users of the system must provide credentials to access the system

5. Log Out
   5.1. The system must allow users to back out to the login screen without needing them to close the application

4 Modeling Requirements
In order to understand program functionality and the generating of various types of reports a UML is shown in Figure 4-1a. The figure is then followed by an english description in Figure 4-1b which is the data dictionary in relation to Figure 4-1b. Figure 4-3 is a use-case diagram that describes the behavior of the system at a high level, followed by an explanation of each use case.
## Audit

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit</td>
<td>Summary of defects on single vehicle</td>
</tr>
</tbody>
</table>

### Attributes

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>plantLocation : enum</td>
<td>Specific GM plant of vehicle being analyzed</td>
</tr>
<tr>
<td>date : date</td>
<td>Date audit was performed</td>
</tr>
<tr>
<td>checkpoint : enum</td>
<td>Checkpoint audit was performed at for defect</td>
</tr>
<tr>
<td>startTime : time</td>
<td>Start time recorded for audit</td>
</tr>
<tr>
<td>endTime : time</td>
<td>End time recorded for audit</td>
</tr>
<tr>
<td>numUnits : int</td>
<td>Total number of cars audited</td>
</tr>
<tr>
<td>DPU : int</td>
<td>Defects per unit of cars</td>
</tr>
</tbody>
</table>

### Relationships

Audits can be accessed through the PDAT System. Audits also have information pertaining to Vehicle.

- **UML Extensions**
  
  QA Reports require audit information. Composition to Defect and Vehicle. Composition from QAReport.

## Defect

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defect</td>
<td>Current defect being recorded</td>
</tr>
</tbody>
</table>

### Attributes

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>location: location</td>
<td>Location on diagram of present defect</td>
</tr>
<tr>
<td>type: enum</td>
<td>Type of present defect (i.e. adhesive, runout)</td>
</tr>
<tr>
<td>severity: enum</td>
<td>Severity of defect, from 1 to 5</td>
</tr>
</tbody>
</table>

### Relationships

Defects are recorded on audits and eventually placed into a QA report

- **UML Extensions**
  
  Composition relationship to audit

## DefectAnalysisReport

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DefectAnalysisReport</td>
<td>A defect analysis report</td>
</tr>
</tbody>
</table>

### Relationships

A defect analysis report may be accessed by QAReports, and a PeriodChart may acess a DefectAnalysisReport. Defect analysis reports also have information pertaining to Vehicle

- **UML Extensions**
  
  A defect analysis report is of type report and therefore derived from Report. Composition to QAReport and Vehicle. Composition from PeriodChart.

## PDAT System

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDAT System</td>
<td>Paint Defect Analysis Tool, application being used to generate reports and input defect data</td>
</tr>
</tbody>
</table>
### Relationships

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PeriodChart</td>
<td>Duration based reports (1 week, 1 month etc)</td>
</tr>
</tbody>
</table>

| Relationships | Audits can be accessed through the PDAT System |

| UML Extensions | A PeriodChart is a type of report and thus derived from Report. Composition to DefectAnalysisReport |

### UML Extensions

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QAReport</td>
<td>Quality Analysis report to be generated</td>
</tr>
</tbody>
</table>

| Relationships | Data from audit may be used in a QAReport. QAReports may also have information useful to a DefectsAnalysisReport. |

| UML Extensions | QAReports are a type of report and therefore derived from Report. Composition to Audit. Composition from DefectsAnalysisReport |

### Element Name

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Report</td>
</tr>
</tbody>
</table>

#### Attributes

| startTime : time | Beginning of time frame that report covers |
| endTime : time   | End of time frame that report covers |
| checkpoint : enum| Checkpoint audit was performed at for defect |

| Relationships | Audits can be accessed through the PDAT System. Reports also have vehicle information. The overall system also may request any report. |

| UML Extensions | Composition to Vehicle. PeriodChart, DefectAnalysisReport, and QAReport are derived types of reports. |

### Element Name

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReportRequest</td>
</tr>
</tbody>
</table>

#### Attributes

| reportType : enum | Beginning of time frame that report covers |

| Relationships | A report request generates a report and is requested by a user |

| UML Extensions | Association with User and Report |

### Element Name

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle</td>
</tr>
</tbody>
</table>

#### Attributes

| make : enum | Manufacturer of vehicle |
The following describes the use cases of the system. The use cases describe the specific event(s) that may occur depending on what the user wants to perform.

**Use Case:** Log In  
**Actors:** Inspector or Analyst (user)  
**Type:** Primary and essential  
**Description:** The user will start the system on their machine and then be prompted to enter their credentials. After they have submitted their login information, the system will authenticate the information and pass the correct users to the menu screen, and it will reject users that provide incorrect login information. Rejected users will not be able to enter any data to the system.  
**Cross Ref.:** 4  
**Use-Cases:** Authenticate User

**Use Case:** Log Out  
**Actors:** Inspector or Analyst (user)  
**Type:** Primary and essential  
**Description:** After the user has submitted their login information, and the system has authenticated the information, they have access to the main menu where they can gather and input data, create audit, and generate reports. At the menu screen, they have the option to securely log out from the system which will prompt the user to enter their credentials.  
**Cross Ref.:** 5  
**Use-Cases:** None

**Use Case:** Authenticate User  
**Actors:** System (Authenticator)  
**Type:** Secondary and essential
**Description:** The user will start the system on their machine and then be prompted to enter their credentials. After they have submitted their login information, the system will authenticate the information and pass the correct users to the menu screen, and it will reject users that provide incorrect login information.

**Cross Ref.:** 4

**Use-Cases:** None

**Use Case: Input Data**

**Actors:** Inspector or Analyst (user)

**Type:** Primary and essential

**Description:** From the menu screen, they have the option to securely log out from the system which will prompt the user to enter their credentials. Once the user has chosen to input data, the system will have the user choose the vehicle to report the defect. After the user has chosen the vehicle, the user will be prompted to enter three data: location, type, and severity of defects.

**Cross Ref.:** 1

**Use-Cases:** Create Audit

**Use Case: Create Audit**

**Actors:** System (Creator)

**Type:** Secondary and essential

**Description:** With the data that the user has entered, the defects will be recorded into the vehicle, and there will be an option for the user to create an audit. This can be compiled later to generate a report.

**Cross Ref.:** 1, 3

**Use-Cases:** None

**Use Case: Generate Report**

**Actors:** Analyst (user)

**Type:** Primary and essential

**Description:** The user will request a report with a time period and report type. The system will gather the data, and then print the file it has produced.

**Cross Ref.:** 2

**Use-Cases:** Gather Data

**Use Case: Gather Data**

**Actors:** System (user)

**Type:** Secondary and essential

**Description:** The system will take the parameters for time period and report type and collect the required data from the database. Also, the system will composite audits from the selected timeframe and generate a report. It will produce a file that will contain the relevant information in a presentable format so that the analyst will be able to determine trends.

**Cross Ref.:** 2

**Use-Cases:** None

5 Prototype

The prototype will include a mobile tablet application for inspectors to record their observations into a database, as well as a web application for supervisors to view and edit records inside of the database. The user interface will contain buttons and menu options for displaying information and details about the inspection.
5.1 How to Run Prototype

In order to operate the prototype, a computer with a modern web browser and mobile tablet is required. The web application is functional on any operating system and does not require any additional plugins. The mobile tablet will operate on both iOS and Android. A Wi-Fi internet connection in the facility where the application will be used is also required. Lastly, a MySQL database is also needed to hold all records.

The prototype can be found at cse.msu.edu/~ahnkevin/prototype.html.

5.2 Sample Scenarios

A sample use case for the prototype can be from a GM automotive plant. Using the mobile tablet application, an inspector would login with their credentials to access the “Model Selection Page” (Figure 4-4a) to begin the inspection process.

Figure 4-4a Mobile Tablet Application “Model Selection Page” Mockup
After selecting GM as the Vehicle Maker, the user will then choose which specific model from GM will be inspected, which in this sample case will be the TRV. After specifying which model the car is, the user will then press the “Begin Inspection” button to begin their analysis of the car. The user is taken to a “Defects Page” (Figure 4-4b) which supplies a menu for the location of the defect, a list of “Add Defect” buttons to record any defects found, and a “Submit Report” button for the user to complete their inspection.

When the user selects one of the “Add Defect” buttons, they will be given a popup textbox so they may fill out any specifications of the defect. In this scenario, the user can add a specification of “chipped paint” to the side of the TRV. After submitting the report, all of the defects noted by the user will be saved into the database.

6 References


7 Point of Contact

For further information regarding this document and project, please contact any member of Group 12. All materials in this document have been sanitized for proprietary data. The students and the instructor gratefully acknowledge the participation of our industrial collaborators.