Functional Specification - David Weir (Student No. 15432642)

Car Damage Analysis

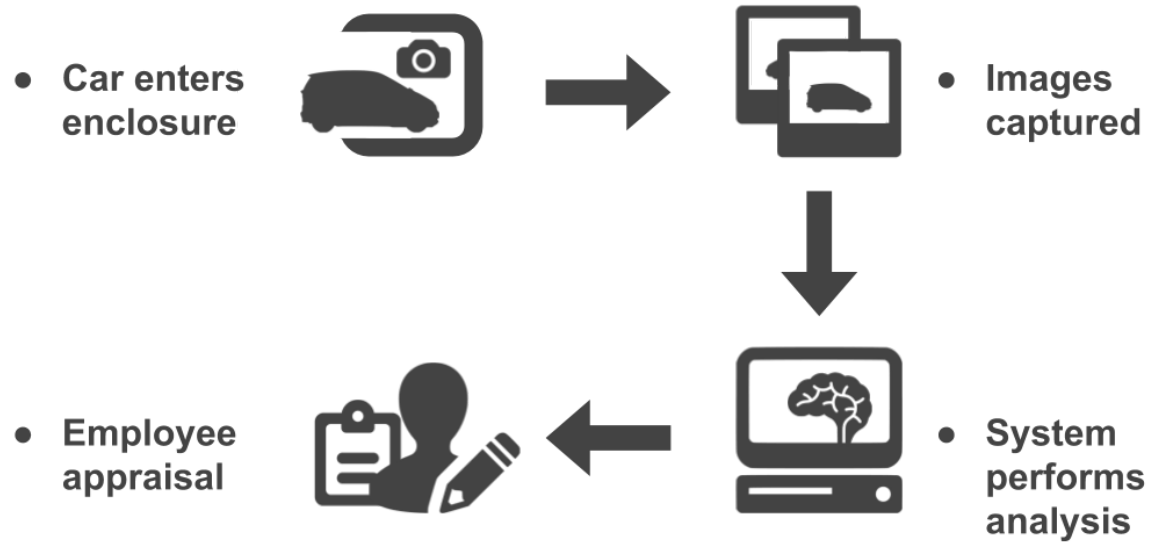
1 Introduction

1.1 Overview

The goal with this system is to automate the process by which cars are returned and inspected for external damages, using the car rental industry as an example but this could be employed by taxi companies, mechanics, or insurance companies. An automated system would provide companies with a more reliable and cost-saving method of vehicle monitoring.

The system will detect when a car has entered the enclosure and take a number of photos of the car from different angles and locations. These images will be sent to a server by the enclosure where a damage analyser will perform a detection on the images to detect damage. Ideally then at this point, the server will send a report to a desktop application highlighting where on the car it has recognised damage. This report can then be used by a company employee to appraise the damage.

Overview as a diagram



1.2 Business Context

This car damage analysis system would be deployed in one of the types of companies mentioned above. It would provide an automated method for damage detection, which could improve reliability and reduce labour costs for the company.

In addition to these benefits for companies, the automatic creation and time stamping of images of cars would allow for easier handling of insurance disputes if a visual history of each car is made searchable by the application.

1.3 Scope

With this project, I hope to develop a minimum viable product (MVP) as soon as possible in order for me to have a system to showcase at the final year demonstrations regardless of the success of the image processing section of this project. This MVP system will display images to employees and also allow them to search the history of a vehicle.

Alongside the development of this MVP, I will create the damage analyser system which will detect the damage in the images of a car. This feature of the project will slot into the previously mentioned application in order to produce the reports that the application will display. The bulk of the work on this project will take place in this section as it is the most unique development challenge to be undertaken.

Finally, after the software portion of the project has been completed, I will develop the hardware enclosure to capture images of a car before and after a rental. The enclosure will send the images of a car to the server to be analysed for damages automatically.

1.4 Glossary

- **Image Processing:** Operations performed on digital images in order to gather useful information and perform analysis.
- **Car or Damage Appraisal:** An employee of a car rental company pricing the damage inflicted on a car and billing the customer appropriately.
- **Appraisal Dispute:** A customer who fights the damage appraisal for a car, due to damage being present before a rental for example.
- **Damage Report:** The output of the damage analysis system. Includes images of the damaged areas on a car. To be used by an employee during the appraisal process.
- **Visual History of a Car:** A list of all previous relevant images for a car, showing previous damages and repair jobs etc.

2 General Description

2.1 Product / System Functions

The main focus of this application will be to develop the damage analyser, which will be able to detect damage in pre- and post-rental images of cars. Ideally this system would be able to spot the differences in two photos of the same car from roughly the same position and highlight areas it deems to be damaged.

Around the damage analyser will be a web application which is used by employees of a car rental company. The application will display the images, both pre and post rental, of a car which the damage analyser has highlighted. This application will also let employees search for cars by registration plate for the purposes of handling insurance disputes.

The final part of the overall system which I hope to develop is the hardware enclosure for the capturing of images of cars. This will be composed of one or several cameras which will automatically take a number of photos of the car from various angles. Once a number of photos have been taken, the enclosure will send these images to a server to be analysed by the damage analyser. This hardware section will most likely only be completed once the software system described above has been developed.

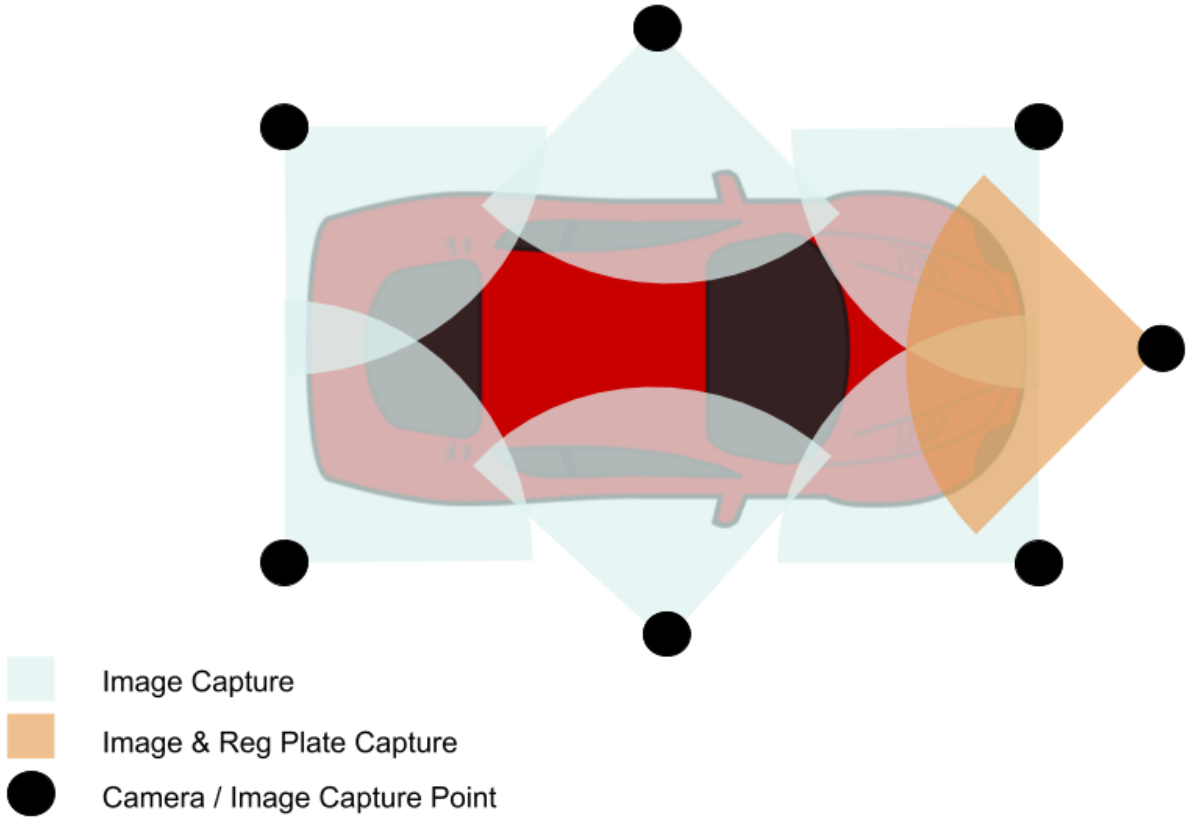
2.2 User Characteristics and Objectives

As the majority of this system is automated, the number of possible users is limited. The main users of this system will be the employees of rental companies who either have to appraise damage made to cars or handle insurance disputes against the rental company. The employee appraising the damage will be provided with the before and after images of the car which they can use to process the appraisal. The employee who handles insurance disputes against the company will make use of the search functionality. The search will then return a visual history of a car based the registration number of the car.

In addition to this, the rental companies' customers who are returning cars are also users in this system, although the extent of their use of the system does not stretch further than driving the car into the enclosure.

2.3 Enclosure Overview

The below diagram provides an overview of the proposed enclosure to capture the images of the car. The diagram shows the number of image capture points of the enclosure and the rough coverage of each of these points.



2.4 Operational Scenarios

The following operational scenarios are outlined in the form of use case descriptions and cover the main functionality of the car damage analysis system from a users point of view. The users in these diagrams are: The employee, the customer and the enclosure itself.

Use Case 1: Damage Appraisal

Goal	Employee Appraises Damages
Scope	Web Application
Actors	*Primary:* Employee *Secondary:* Enclosure
Success Conditions	Employee successfully appraises a damaged car
Preconditions	Enclosure has run detection on the images of the car.
Description	1. Employee enters web application 2. A list of cars awaiting appraisal is displayed 3. Enters page for a car appraisal 4. Web application displays the images of the car before & after a rental 5. Employee uses the images provided by the application during the appraisal
Success Outcome	Employee successfully appraises the car
Failure Outcome	Employee cannot appraise damage due to a false positive recognition from the damage analyser

Use Case 2: Appraisal Dispute

Goal	Employee handles a customer dispute of damage
Scope	Car Search Functionality
Actors	*Primary:* Employee *Secondary:* Customer
Success Conditions	Employee is provided with historical information of car
Preconditions	Enclosure has detected damages; An employee has appraised the damages on the car.
Description	1. Customer disputes employees appraisal 2. Employee enters web application 3. Employee searches for car by vehicle registration number 4. Web application displays a visual history of a car (A list of previously taken images)
Success Outcome	The employee handles the customers dispute using the images
Failure Outcome	The application fails to display historical information due to: A. No historical information for a car B. A mistyped search query

Use Case 3: Damage Analysis

Goal	To analyse a car for damages
Scope	Damage Analyser
Actors	*Primary:* Enclosure *Secondary:* Customer
Success Conditions	Car is successfully analysed for damages
Preconditions	Images have been taken of the car before and after the rental
Description	1. Customer enters enclosure 2. Enclosure recognises car license plate 3. Enclosure captures images of the car from various angles 4. The images are sent to a server to be analysed
Success Outcome	Car is successfully analysed for damages
Failure Outcome	Car cannot be analysed for damages due to connectivity issues with the server. The enclosure will then save the images locally in order to send them to the server when it reconnects.

Use Case 4: Report Flagging

Goal	Employee flagging incorrect damage report
Scope	Web Application
Actors	*Primary:* Employee
Success Conditions	Employee can flag an analysis as incorrect
Preconditions	Damage analyser has incorrectly reported a car as damaged
Description	1. Employee enters web application 2. Employee notices a false positive result in the web application 3. Employee chooses to flag the images as incorrect 4. Images are stored as incorrect in the database
Success Outcome	Incorrect images are flagged in the database to be used to tune the analyser in the future
Failure Outcome	N/A

Use Case 5: Returning a Car

Goal	Customer returns a car
Scope	Enclosure
Actors	*Primary:* Customer *Secondary:* Enclosure
Success Conditions	Customer's car is appraised for damages
Preconditions	Images have been taken of the car before the customers rental
Description	1. Customer enters the enclosure 2. Enclosure captures images of the car from various angles 3. Enclosure sends the images to be analysed by the damage analyser on the server. 4. Damage analyser completes the analysis 5. Damage analyser sends a notification of the analysis outcome to the enclosure and customer.
Success Outcome	Customer's car is analysed successfully

2.4 Constraints

- **Demonstration:** Due to the large nature of the hardware for this project, it is possible that a scale model of the hardware will have to be developed in order for this portion of the project to be demonstrated in the DCU computing labs.
- **Speed:** The speed of the system will need to be fast enough such that a rental car customer does not become frustrated waiting for the result of their inspection.
- **Learning:** Quite a bit of time will be dedicated to learning how to perform image processing well as I have never dealt with this before.
- **Time:** There are a number of large sections to this project, all of which must be completed by the project deadline of 19 May 2019.

3 Possible Implementations

3.1 Computer Vision Algorithms

The first possible solution to the problem I will be attempting to implement will revolve around the use of feature detection algorithms to highlight possible differences in the images of a car. I will attempt to make use of the “SURF” algorithm for feature detection to highlight any new features in the after photo from a car rental. These features can then be compared to stored images of damage to detect whether these new features resemble damage or not.

Advantages:

- Less training data and computational power is required.
- Detect differences in before and after photos, this allows for damage present pre-rental to not be highlighted when analysing the post-rental images.

Disadvantages:

- No known algorithms for detecting differences in two images of the same object have been found at this time.

3.2 Object Detection Using Deep Learning

The next possible solution involves the use of a deep learning model to detect whether damage is present in the “after” photo of a car rental. The model would be trained on a large dataset of different types of damage to cars such as scratches and dents. The model will return a bounding box surrounding the area that it recognises damages.

Advantages:

- Similar projects have been completed in the past, proving that it is feasible

Disadvantages:

- Cannot detect differences by comparing before and after photos. It can only detect damaged elements in the post rental photo of a car.
- A large dataset of damage to cars needs to be acquired.

4 Functional Requirements

4.1 Damage Analysis on Images of a Car

Description:
The analyser must receive images and compare these images with previously taken images to highlight areas of difference that could be damaged.
Criticality: Pivotal as this functionality makes up the main component of the system.
Technical Issues:
Very few (if any) well known algorithms or methods for detecting the differences between two photos of the same object have been found.
Dependencies:

- 4.3: This functionality benefits from the ability of the enclosure to automatically capture images to be analysed.
- 4.4: The images must be transferred to the server in order to be analysed for damages

4.2 Displaying Images of Cars Pre & Post Rental

Description:
The web application must be capable of displaying the images of cars from before and after an rental in order for employees to appraise damages, handle disputes and flag false positive results.
Criticality: Very important as this features allows the employees to view the highlighted damages before ever going outside to view the cars in person. This feature would be an important cost saving aspect of the project for car rental companies.
Technical Issues:
Storing images could lead to challenges in dealing with hard drive space. Displaying these images efficiently could also present some difficulty.
Dependencies: N/A

4.3 Automatic Image Capture

Description:
The enclosure must recognise the registration plate of a car entering the enclosure and capture images from various angles around a car.
Criticality: An important aspect of the project as it will aid in the automation of the whole process involved in the return of a car, though its is not vital for the completion of the bulk of this project.
Technical Issues:
Issues could arise with having a constant stream of images passing to the reg plate analyser from the cameras. Measures might need to be in place in order to ensure the full car is present in the photos taken by the enclosure. I have very little experience with programming for hardware, some new challenges could arise from this.
Dependencies: N/A
External Dependency: License plate detector must be run on the controller of the enclosure reliably.

4.4 Image Transfer to Damage Analyser

Description:
Images must be quickly and reliably transferred between the enclosure and the server, in order for the system to process a customer with speed.
Criticality: As with the last requirement, this requirement is necessary in order to automate the process of returning a car.
Technical Issues:
Compression algorithms will need to be employed when transferring the images in order to decrease the time it takes for analysis on a car to complete.
Dependencies:

- 4.3: This requirement relies on the enclosure to automatically capture images in order to be transferred to the server for analysis.

4.5 Searchable Car History

Description:
Historical images of each car must be searchable and viewable for the purposes of handling any damage appraisal disputes that may arise. These images must be searchable by license plate.
Criticality: This requirement is not as important for the main functionality of the system but the inclusion of the search functionality would ease the workload for employees who handle disputes for the car rental companies, which is one of the main benefits of this project.
Technical Issues:
Images will need to be reliably stored and timestamped in such a manner as to make them easily searchable.
Dependencies:

- 4.2: This feature depends on requirement 4.2 as the search results, containing all previous images for a car, must be displayed to the user.

4.6 Flagging a damage analysis report as incorrect

Description:
In order to improve this system over its lifespan, it is necessary that an incorrect analysis report is flagged as incorrect. This image can then be used at a later date to improve the reliability of the damage analyser.
Criticality: While not critically important for the completion of this project, the addition of a flagging functionality would help improve reliability over the lifespan of the system.
Technical Issues:
Again, images will have to be reliably stored and timestamped in order to be retrieved at a later date.
Dependencies:

- 4.1: In order for an image to be flagged, it needs to first be analysed for damages incorrectly.

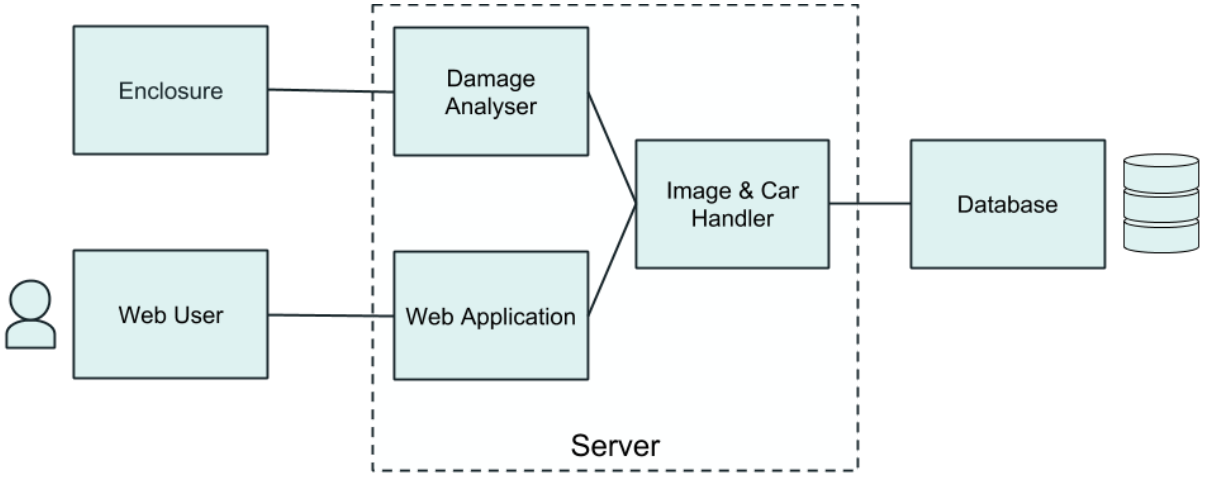
4.7 Account Sign in

Description:
Employees must be able to sign into the application using a Google account. This Google account is a placeholder for what would be the employees official account within the rental company.
Criticality: Not hugely important as it is simply a placeholder for the official company account, but would be beneficial for the purposes of demonstrations.
Technical Issues:
The Google account refresh and access tokens will need to be handled as they can update every few hours. This could present a challenge when authorising users in the web app.
Dependencies: N/A

5 System Architecture

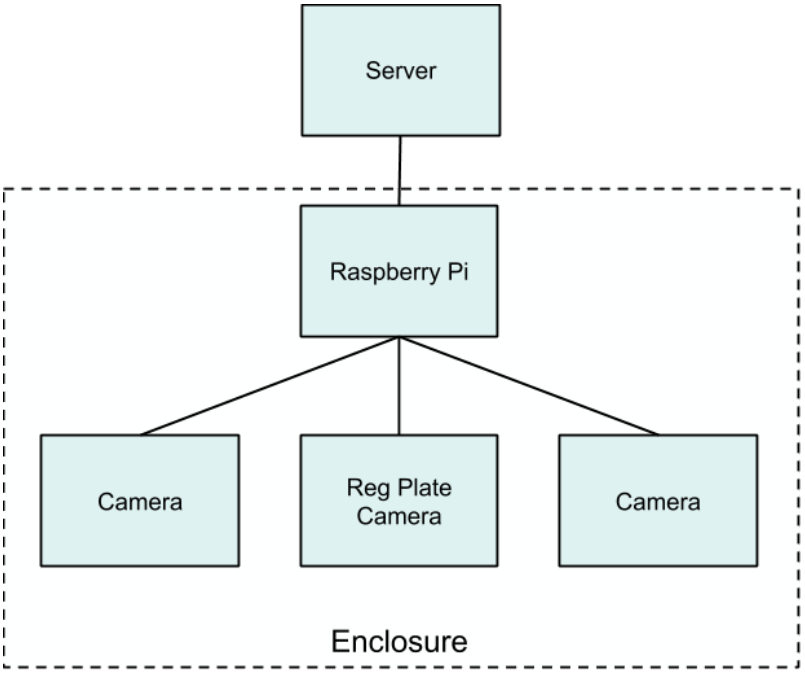
5.1 Software System

The software system covers the proposed sections to be implemented for the actual damage analysis and image display portions of the overall system.



5.2 Hardware System

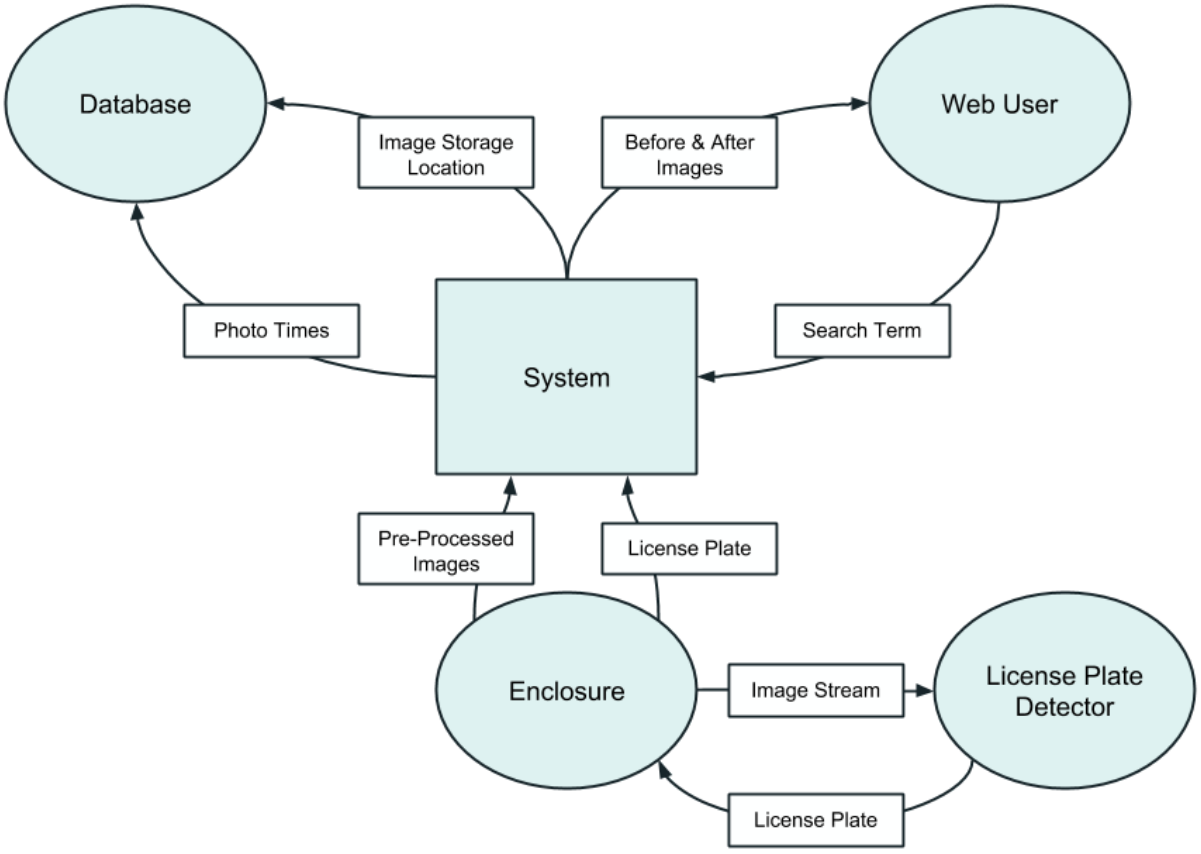
The hardware system covers the proposed sections to be implemented for the image capturing portion of the overall system.



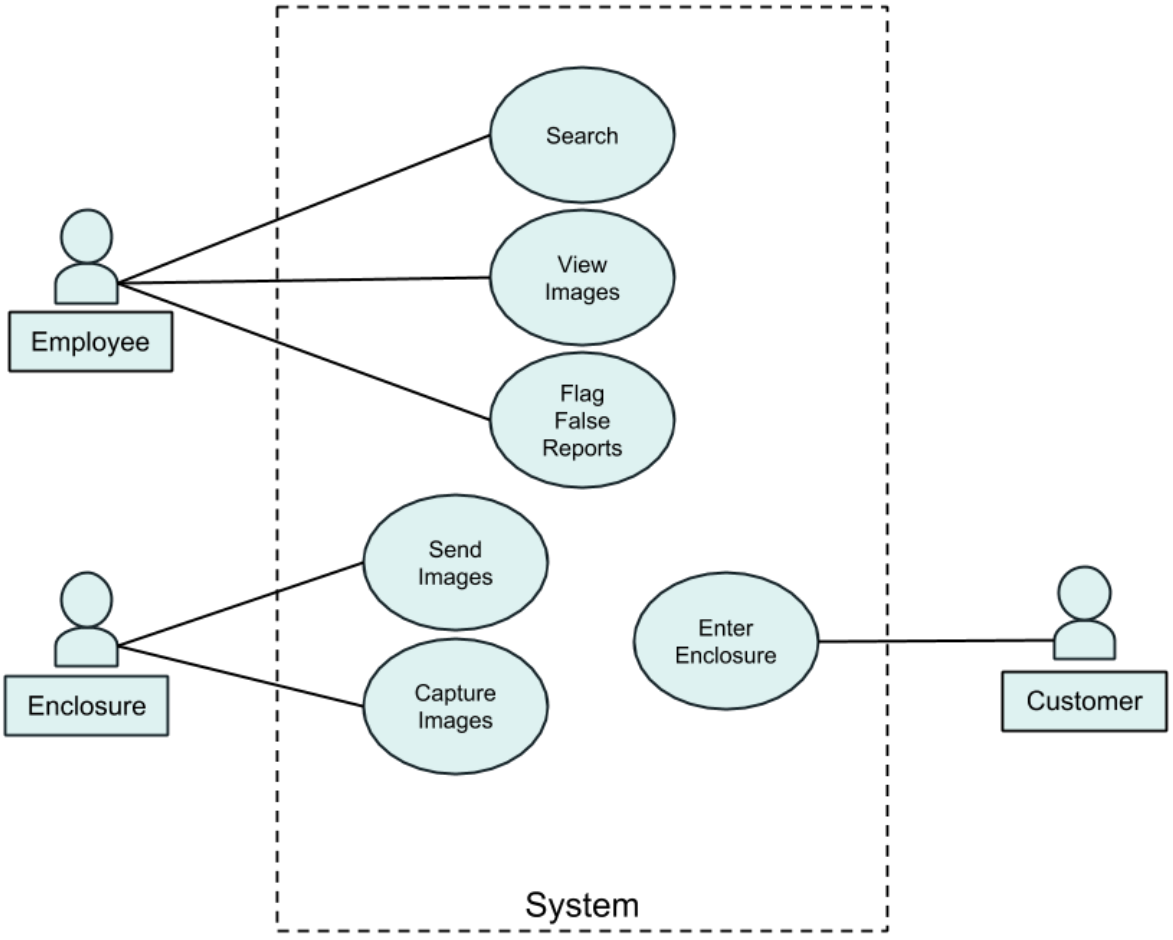
6 High-Level Design

6.1 Context Diagram

In this context diagram, the “System” is viewed as the software system that will be performing the damage analysis on the images, while the enclosure is viewed to be outside the boundary of the software system.

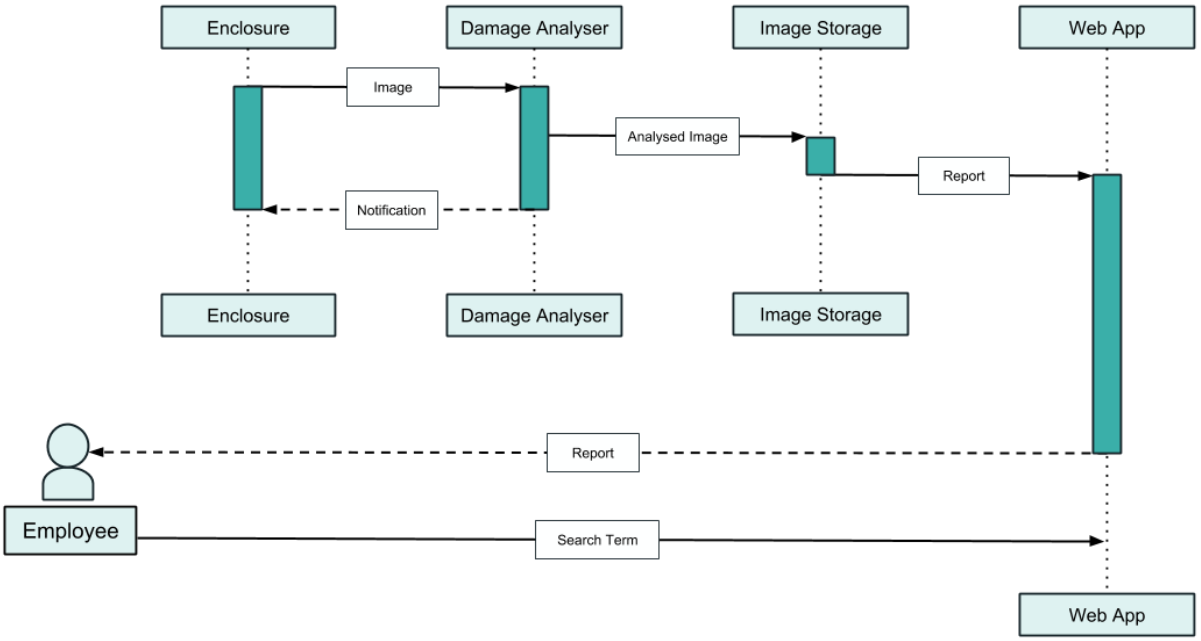


6.2 User Diagram

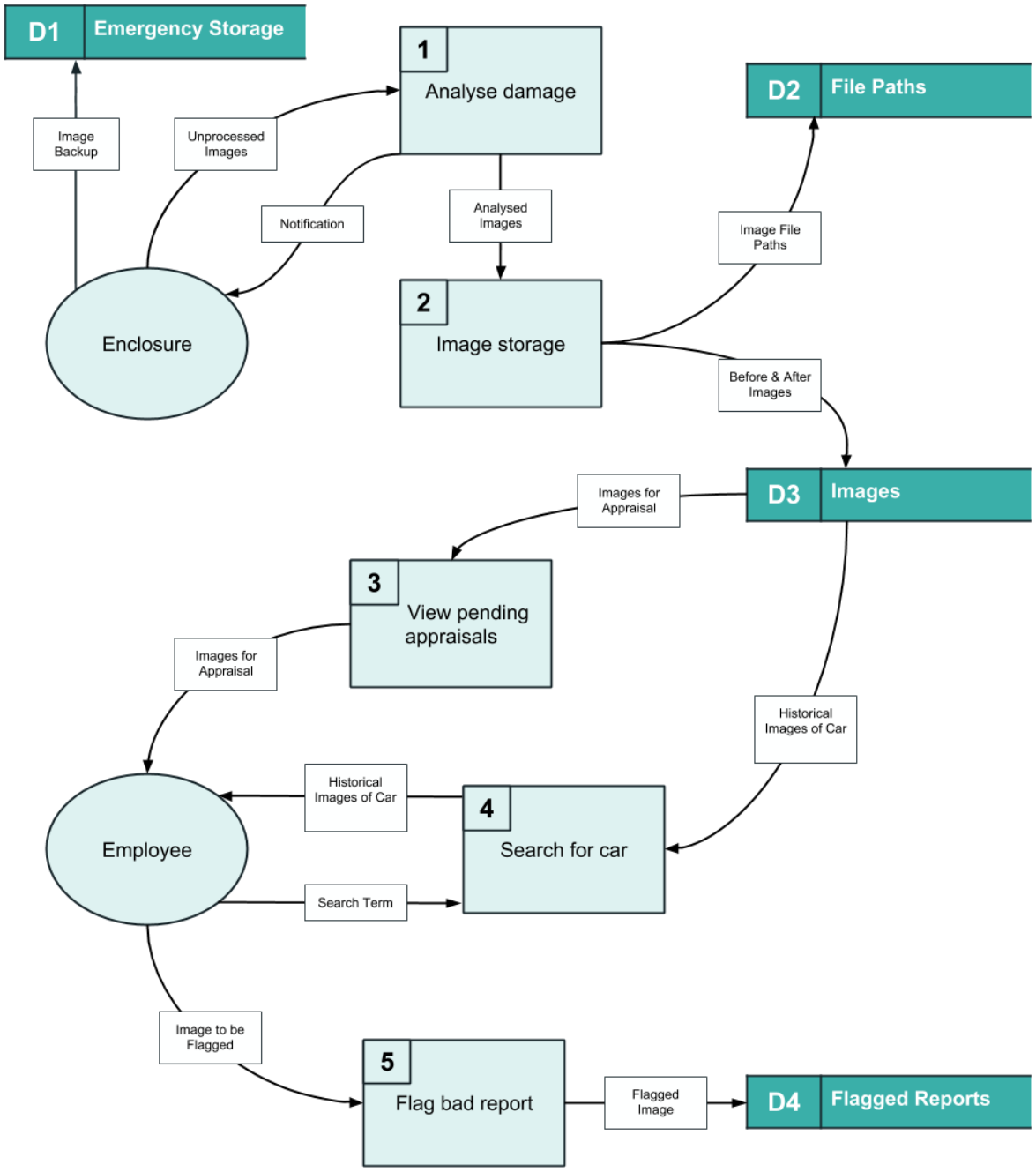


6.3 Sequence Diagram

Below is a sequence diagram which covers roughly the path an image may take through the overall system. It provides a walkthrough how an image is transferred between the different components of the system, from capture to damage analysis and finally, to its display as part of a report in the web app for an employee.

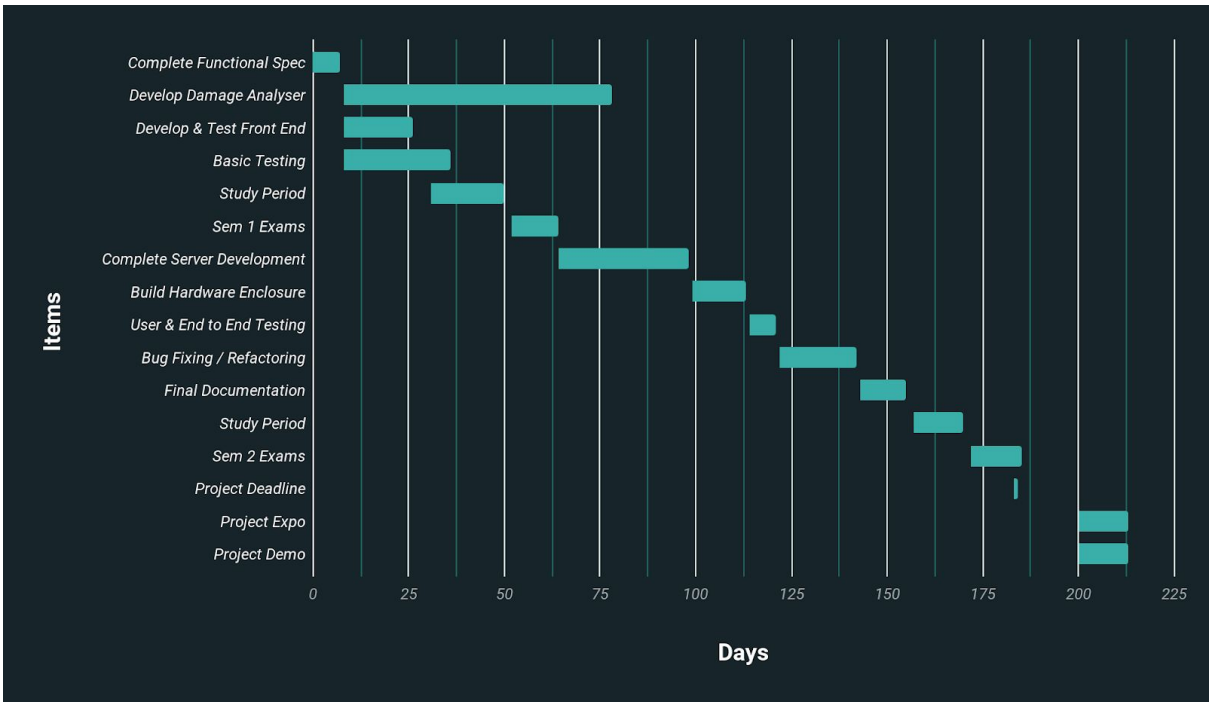


6.4 Data Flow Diagram



7 Preliminary Schedule

7.1 Gantt Chart



7.2 Key Dates

Activity	Start Data	End Date
Complete Functional Spec	16/11/2018	23/11/2018
Develop Damage Analyser	24/11/2018	02/02/2019
Develop & Test Front End	24/11/2018	12/12/2018
Basic Testing	24/11/2018	22/12/2018
Study Period	17/12/2018	05/01/2019
Sem 1 Exams	07/01/2019	19/01/2019
Complete Server Development	19/01/2019	22/02/2019
Build Hardware Enclosure	23/02/2019	09/03/2019
User & End to End Testing	10/03/2019	17/03/2019
Bug Fixing / Refactoring	18/03/2019	07/04/2019
Final Documentation	08/04/2019	20/04/2019
Study Period	22/04/2019	05/05/2019
Sem 2 Exams	07/05/2019	20/05/2019
Project Deadline	18/05/2019	19/05/2019
Project Expo	04/06/2019	17/06/2019
Project Demo	04/06/2019	17/06/2019