

USED CAR VALUE PREDICTOR

Group 16



MAY 03, 2020 SUBMISSION BY:

Zoya Ahmad, Amish Singh, Rohan Swain, Nitin Singh

TABLE OF CONTENTS

S. No.	DATE	CONTENT	Page No.	SIGNATURE
1.		Problem Statements a) Identify the Software Project b) Create Business Case c) Arrive at a Problem Statement	5	
2.		Stakeholders & Process Models a) Stakeholder and User Description • Identifying Stakeholders • User story b) Identify the Appropriate Process Model c) Arrive at a Problem Statement d) Comparative Study with Agile	10	
3.		Identifying the Requirements from Problem Statements a) Requirements Elicitation b) Feasibility Study c) Functional Requirements d) Non-Functional Requirements e) System Requirements	14	
4.		Project Plan a) Project Plan b) Identify Job Roles and Responsibilities	17	
5.		Project Effort Based on Resources a) Work Breakdown Structure b) Risk Analysis	24	
6.		Estimation of Project Metrics Function Point Analysis COCOMO Model	33	
7.		Design System Architecture	36	
8.		Modeling UML Use Case Diagrams and Capturing Use Case Scenarios a) Use case diagrams • Identifying Actors • Use Case b) Association between Actors and Use Cases c) Use Case Relationships • Include Relationship • Extend Relationship d) Generalization Relationship	38	
9.		E-R Modeling from the Problem Statements	40	

T		Г	
	a) Entity Relationship Model		
	b) Entity Set and Relationship Set		
	c) Attributes of Entity		
	 Keys 		
	 Weak Entity 		
10.	Identifying Domain Classes from the		
	Problem Statements		
	a) Domain Class	42	
	b) Using Generalization		
	c) Using Subclasses		
11.	Statechart and Activity Modeling		
	a) Statechart Diagrams		
	State		
	Transition		
	 Action 	43	
	b) Activity Diagrams		
	Components of an Activity		
	Diagram		
12.	Modeling UML Class Diagrams and		
12.	Sequence diagrams		
	a) Class diagram		
	Class Class	46	
		40	
	Relationships		
	b) Sequence diagram		
	Object		
40	c) Life-line bar		
13.	Modeling Data Flow Diagrams		
	a) Data Flow Diagram	40	
	b) Context diagram and leveling	49	
4.4	DFD		
14.	Implementation		
	a) Addition of New requirement	51	
	b) Addition of New Issue		
15.	Estimation of Test Coverage Metrics		
	and Structural Complexity		
	a) Control Flow Graph		
	b) McCabe's Cyclomatic Complexity	52	
	c) Optimum Value of Cyclomatic		
	d) Complexity		
16.	Designing Test Suites		
	a) Software Testing		
	b) Testing Frameworks	54	
	c) Master Test Plan		
	d) Manual Testing		
17.	Deployment Report		
18.	Conclusion	56	
10	References	57	
19.	VEIGIGIICE2	57	

PROJECT COMPLETED IN THE SESSION 2019-2020

TITLE / ROLE

Used Car Value Predictor



PROJECT

- It is a Machine Learning model
- Model predicts the resale value of used cars
- Regression analysis is incorporated for prediction
- Model learns from the historical data on used car transaction

THE HISTORY

Earlier car's resale value was intuitively quoted by experts or dealers based on their knowledge and mechanical evaluations. The final price was finally landed upon after bargaining between the two sides.

A regression model instead provides a mathematical and a uniform way to quote a price by regression analysis on the historical sales data. It offers a clear way to demonstrate the depreciation of a car's value.

LIMITATIONS

- If a car belonging to a newly established brand or a new-in-the-market brand is put up for sale, then the unavailability of the past sales data will cause the model to give no / incorrect results.
- Users have to provide history details of the car; if some details are not available with the user, it may create a problem

APPROACH

- Data collection and exploration
- Machine Learning model selection
- Training and testing the model
- Making prediction
- Accuracy test

BENEFITS

- 1. Brings uniformity in the used-vehicle market
- 2. Transparency between the transacting parties
- 3. Save the dealers from convincing the buyers for the quoted price
- 4. Help sellers know the price that their cars may yield
- 5. Provides appropriate and accurate cost
- 6. Help dealers to device specific sales strategies for the cars in the inventory

Project Name	Used	d Car Value Pr	redictor
Project Sponsor		Project Manager	
Date of Project Approval		Last Revision Date	
Contribution to Business Strategy	The regression approach of the model can reshape the existing evaluation system of the used car market which was originally intuition based owing to the knowledge of the experts, dealers and mechanics. That implied the lack of standardization and uniformity. Model will bring uniformity on the mathematical grounds, using past sales data and quoting a value which is convincingly transparent enough for both the transacting parties		
Options Considered	Options considered included: • For an enhanced user experience, fabrication of a web app for the same. • For expansion of customer base, training the model on data from multiple countries		
Benefits	 Brings uniformity in the used-vehicle market Transparency between the transacting parties Save the dealers from convincing the buyers for the quoted price Help sellers know the price that their cars may yield Provides appropriate and accurate cost Help dealers to device specific sales strategies for the cars in the inventory 		
Risks	 If a car belonging to a newly established brand or a new-in-the-market brand is put up for sale, then the unavailability of the past sales data will cause the model to give no / incorrect results. Users have to provide history details of the car; if some details are not available with the user, it may quote an incorrect price. 		

PROJECT PROBLEM STATEMENT

1. PROJECT DESCRIPTION

The project is a Machine Learned model that incorporates the regression analysis methodology for estimating the resale value of the car.

The model is trained and tested on the available voluminous historical data on car sales that mainly includes almost all the major brand cars in the market.

The model takes into account all the attributes of a used car as its 'features' (eg. Mileage, condition, etc.) for analysis.

Framework chosen for the realisation of the model is Python and its associated data exploration and manipulation libraries.

2. PROJECT CONSTRAINTS

1. What are the needs and purposes for the work?

- Brings uniformity in the used-vehicle market
- Transparency between the transacting parties
- Save the dealers from convincing the buyers for the quoted price
- Help sellers know the price that their cars may yield
- Provides appropriate cost

2. What key issues should be considered?

- Missing values of the 'features' in the past sales dataset.
- If a car belonging to a newly established brand or a new-in-the-market brand is put up for sale, then the unavailability of the past sales data will cause the model to give no / incorrect results.
- At the user's end, the user needs to provide history details of the car; if some details are not available with the user, it may result in incorrect evaluation of the car.

3. What are the goals and objectives of the work?

- To brings uniformity in the used-vehicle market
- Standardisation of the used car evaluation system
- To bring the transparency between the transacting parties
- To help the market lock a deal where both the parties are satisfied, incurring no loss to the either side.
- To expedite the market dealings by cutting on negotiations and bargains.

4. Who is the audience?

- Online cab service companies, who can utilise it for maintaining their inventory
- Online car dealers, who can give out this model as a part of their web service in the automobile industry

- Sellers, who can utilise it for price check purpose.
- Local town dealers, who can use it to device specific sales strategies for the cars in the inventory

5. What types of usable information and tools are available for practical?

- Software Requirements:
 - Windows 10 operating system
 - Python install version 3.8.1
 - Integrated Development Environment such as Spyder, Anaconda
 - Pip install component for importing libraries (Pandas, Numpy, Scikit-learn)
- Hardware Components:
 - Processor intel i7
 - Hard Disk >500GB
 - Memory 8GB RAM
- Additional Components:
 - Dataset of the historical sales data

3. SCHEDULE, RESOURCE AND BUDGET CONSTRAINTS

1. What are the Existing Resources? Are they internal or external?

Resources at disposal for the project include:

- Specialised integrated development environment (Anaconda)
- Car sales data
- Data exploration/visualisation tools
- Data manipulation and machine learning integrated libraries
- Local personal computer powerful enough for processing the stated model

The resources are both internal as well as external.

2. What are the time constraints that may dictate the delivery of work items?

Quality of the dataset—one major delivery time binding constraint is determined by the condition of the collected dataset. A Dataset with alot of missing or incorrect values will require more effort and time for the normalisation task than the one with lesser faulty data points.

Tuning of the model—feature engineering might have to be incorporated to create features for the model to train on and give enhanced outcomes.

3. What is the availability and quality of existing data?

Ample types and volumes of ready data is available on the web from various number of sources, that also includes car sales data from almost all the countries of the world, making it possible to train the model for any given setting or custom requirements. However data is raw enough to require cleaning and maybe feature engineering.

BUSINESS CASE TEMPLATE

DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038) Zoya Ahmad (RA1811027020028)
	Rohan Swain (RA1811027020051) Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



THE PROJECT

In bullet points, describe the problem this project aims to solve or the opportunity it aims to develop.

- It is a Machine Learning model
- Model predicts the resale value of used cars
- Regression analysis is incorporated for prediction
- Model learns from the historical data on used car transaction

THE HISTORY

In bullet points, describe the current situation.

- Earlier car's resale value was intuitively quoted by experts or dealers based on their knowledge and mechanical evaluations.
- The final price was finally landed upon after bargaining between the two sides.
- A regression model instead provides a mathematical and a uniform way to quote a price by regression analysis on the historical sales data.
- It offers a clear way to demonstrate the depreciation of a car's value.

LIMITATIONS

List what could prevent the success of the project, such as the need for expensive equipment, bad weather, lack of special training, etc.

- If a car belonging to a newly established brand or a new-in-the-market brand is put
 up for sale, then the unavailability of the past sales data will cause the model to give
 no / incorrect results.
- User must provide required history details of the car; if some details are not available with the user, it may quote an incorrect price.

APPROACH

List what is needed to complete the project.

- Data collection and exploration
- Machine Learning model selection
- Training and testing the model
- Making prediction and visualisation
- Accuracy test

BENEFITS

In bullet points, list the benefits that this project will bring to the organization.

- Brings uniformity in the used-vehicle market
- Transparency between the transacting parties
- Save the dealers from convincing the buyers for the quoted price
- Help sellers know the price that their cars may yield
- Help dealers to device specific sales strategies for the cars in the inventory

DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038) Zoya Ahmad (RA1811027020028) Rohan Swain (RA1811027020051) Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



STAKEHOLDERS & PROCESS MODELS

Stakeholders are those people who can affect a project positively or negatively.It's exactly what the word suggests; if someone's got a stake in the project, they are a stakeholder.

They comprise of suppliers, internal staff, members, customers (shareholders, investors, consumers, etc.), regulators, local and regional communities etc; they can be anyone who has a stake in the company's success for a variety of reasons. Additionally, stakeholders may include purchasers, clients, owners, and non-governmental organizations. Sponsors, customers, dealers, buyers, sellers, etc. are all considered to be stakeholders.

Various stakeholders of this project include—

- 1. Model developers
- 2. Project manager
- 3. Individual sellers
- 4. Dealers
- 5. Automobile brands
- 6. Online buy and sell companies

USER DESCRIPTION

1. User Story Example 1

As an owner of a car to be sold I can utilise the used car value predictor So that I can know the price that my car may yield.

Acceptance Criteria Examples

- The brand and the mode of the car is selected.
- The details of the chosen car are entered
- The value of the car is evaluated
- ♣ The market value of the car is displayed

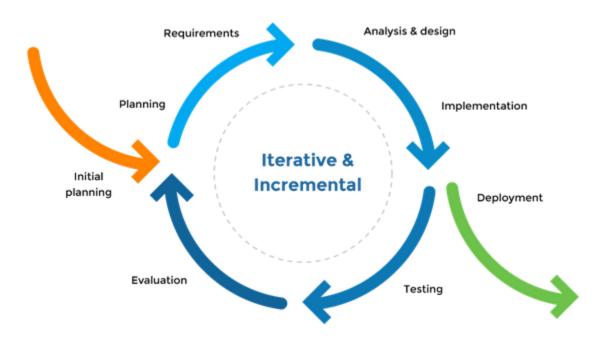
2. User Story Example 2

As a used car dealer I can utilise the used car value predictor So that I can know the price to be quoted before the seller

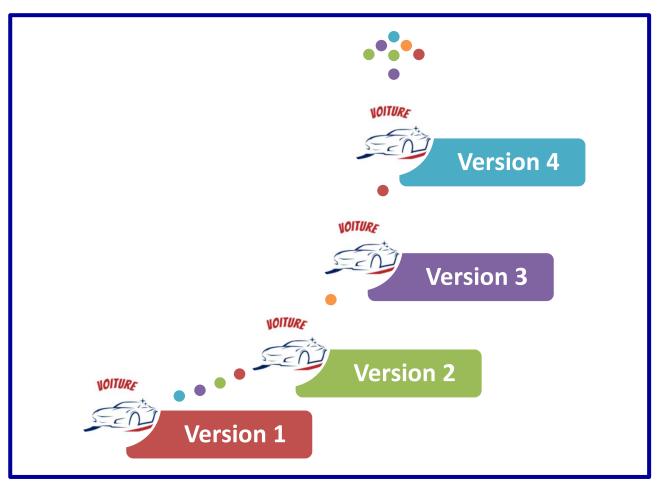
Acceptance Criteria Examples

- The brand and model of the brought in car is selectedThe details of the chosen car are entered
- ♣ The value of the car is evaluated
- ♣ The price to be quoted is displayed

PROCESS MODEL—



Iterative and Incremental Model



PROBLEM STATEMENT

The **Used Car Value Predictor** is a Machine Learned model that incorporates the regression analysis methodology for estimating the resale value of the car.

The model is trained and tested on the available voluminous historical data on car sales that mainly includes almost all the major brand cars in the market.

The model takes into account all the attributes of a used car as its 'features' (eg. Mileage, condition, etc.) for analysis.

Framework chosen for the realisation of the model is Python and its associated data exploration and manipulation libraries.

COMPARISON BETWEEN AGILE AND PROCESS MODEL

AGILE MODEL

INCREMENT MODEL

Agile model is an incremental delivery process where each incremental delivered part is developed through an iteration after each time box. The main principle of the Agile model is to achieve agility by removing unnecessary activities that waste time and effort.

The requirements of the software are divided into several modules that can be incrementally developed and delivered. The core features are developed first and the whole software is developed by adding new features in successive versions.

In the Agile model, end date for an iteration is fixed, it cannot be changed. The development team may have to decide to reduce the delivered functionality to complete that iteration on time.

In the Incremental development model, there is no fixed time to complete the next iteration.

DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038)
	Zoya Ahmad (RA1811027020028)
	Rohan Swain (RA1811027020051)
	Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



REQUIREMENTS FROM PROBLEM STATEMENT

REQUIREMENTS

The model is capable of clearing some of the dirt that sticks around this used car sales industry. Majority of the deals that ever take place, certainly leave either of the transacting sides unsatisfied or in dilemma doubting the worth of the deal that took place. It is hard to find a seller who is satisfied with the first price quotation given by the dealer; or a dealer who stands satisfied at the negotiation price demanded by the seller. This whole wave of dissatisfaction and chaos arises due to the fact of lacking standardization of any sort.

At the broader level, the model holds the potential of serving the used car industry in several ways, with bringing about a mathematical basis of standardization of the whole dealing process.

USER REQUIREMENTS

- The usability of the model is not just limited to the sellers and dealers, but it has also
 got those agencies/businesses covered that aim at keeping a fleet of vehicles for
 strategically monitoring and maintain the inventory.
- The model quotes the approximate cost that the owner should be entitled with in the market upon selling his/her car; given the parameters judging the car's condition. (A seller's perspective of potential usage)
- Similarly, it can quote the approximate cost that the dealer should entitle the customer within exchange for his/her car; given the parameters judging the car's condition. (A dealer's perspective of potential usage)

Businesses running cab services can buy cars to include in its fleet, governed by the
potential cost that they might yield after a fixed year of usage.

SYSTEM REQUIREMENTS

- Software Requirements—
 - Operating System (OS) such as Windows, Linux, etc.
 - Python Programming Language (Domain-specific)
 - Visual Studio Code 15.0
- Hardware Components—
 - Processor: Intel or AMD
 - Hard Disk
 - Memory: minimum 4GB RAM

FUNCTIONAL REQUIREMENTS

PRODUCTS:

- User must be able to give in the required inputs for the requested data fields, representing the corresponding "feature" values for the model to perform the regression on.
- For the ease of input, selection of the desired value as one of the options from a drop-down menu for specific fields with obvious finite options. Example, age of the car, car's brand etc.
- Upon filling the details of the required car model, the system displays a thorough product description consisting of the brand, make, model, mileage, handling and suspension and safety measures and equipment
- The user has an option to compare the prices offered by different dealers; the user can then select the product which best meets his/her requirements
- The model should be able to perform regression on all the feature variables pertaining to the entered brand (of the car) only.
- For the fields, whose values are unknown to the user, the model shall implicitly tag it with the mean value from all the data points for the chosen brand.
- Upon the performance of the back-end regression operation, the output shall be made available at the user's end.

PAYMENTS:

The customer can book an appointment/meeting with the dealer by paying a small convenience fee

- The system shall accept all major payment methods like credit cards, debit cards, net banking, wallets, etc.
- ♣ The system shall validate payment with the capability of the model to predict the car's value is limited to those brands only, on whose data the model is trained on.
- A car belonging to the foreign brand or a newer brand shall stand as a limitation to the model
- Guessed or implicitly approximated values of the features shall lead to compromised accuracy of the generated results.

NON-FUNCTIONAL REQUIREMENTS

CAPACITY

- ♣ The system shall support around 100 transactions per day
- ♣ The system shall support a peak transaction rate of 5 transactions per second
- ♣ The system should allow access to a minimum of 500 users in concurrent sessions

AVAILABILITY

- ♣ The system shall be available to the users 24x7
- ♣ The system shall accept the convenience fee and arrange an appointment with the dealer within 5-10 seconds in 90% of the cases
- ♣ The system shall allow a user login within 5-10 seconds

DOMAIN REQUIREMENTS

- The prices shall be evaluated as per the prevalent market standards of the country.
- The data entered by the user shall remain confidential under the bounds of the Data Protection Act.
- The cost depreciation rate shall be accounted on the basis that complies with the market standards to date.

DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038)
	Zoya Ahmad (RA1811027020028)
	Rohan Swain (RA1811027020051) Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



PROJECT PLAN

1. Presenting the Project Sketch before Stakeholders

- The model is capable of clearing some of the dirt out, that sticks around this used car sale industry; Majority of the deals that ever take place, certainly leave either of the transacting sides unsatisfied or in dilemma doubting the worth of the deal that took place.
- The project is a Machine Learned model that incorporates the regression analysis methodology for estimating the resale value of the car.
- The model is trained and tested on the available voluminous historical data on car sales, that mainly includes almost all the major brand cars in the market.
- The model takes into account all the attributes of a used car as its 'features' (eg. Mileage, condition etc) for analysis.
- The framework was chosen for the realization of the model in Python and its associated data exploration and manipulation libraries.
- It holds the capability of helping petty town dealers of individuals to established businesses to serve even better.

2. Roles and Responsibilities

Project Manager

- Develop a project plan
- Manage deliverables according to the plan
- Recruit project staff
- Lead and manage the project team
- Determine the methodology used on the project
- Establish a project schedule and determine each phase

Assign tasks to project team members

Data Analyst

- Responsible for the accusation of the data
- Performs data cleansing
- Analyze the quality of data
- Analyze the trends and relations among various features
- Perform feature engineering if needed

Developer

- The backbone that leads to the realization of the implementation aspect
- Writes the code to test and train the model
- Performs project design and development activities according to customer specifications
- Works with manager in preparing project plan, budget and schedule
- Tracks progress regularly and develop status reports to management
- Develops cost reduction initiatives while maintaining quality and productivity

Project Team Member

- Contributing to overall project objectives
- Completing individual deliverables
- Providing expertise
- Working with users to establish and meet business needs
- Documenting the process

Project Sponsor

- Make key business decisions for the project
- Approve the project budget
- Ensure availability of resources
- Communicate the project's goals throughout the organization
- Provide direction and guidance for project empowerment, key business strategies and project initiatives
- Participate in initial project planning, including developing the project chart and the project scope
- Evaluate the project's success on completion

Business Analyst

- Assist in defining the project
- Gather requirements from business units or users
- Document technical and business requirements
- Verify that project deliverables meet the requirements

Test solutions to validate objectives

3. Kickoff Meeting

- ♣ A kickoff meeting is an essential tool to communicate with stakeholders. If done correctly, it can help to execute the project with minimal obstruction, and one can finish the project with fewer obstacles.
- ♣ A kickoff meeting is one of the first meetings held amongst the project stakeholders when starting a new project.
- ♣ This meeting includes the high-level project stakeholders such as the project sponsor, management, and the project manager, as well as the team members and the project manager.
- ♣ This meeting plays a vital role because it gives a project manager the opportunity to define the common goal and the purpose of completing the project and as well build a consensus.

4. Scope Statement

- A project scope, or project scope statement, is a tool used to describe the major deliverables of a project including the key milestones, high-level requirements, assumptions, and constraints.
- A project scope statement is a useful tool for future decision making when new change requests are considered to modify the project scope.
- An effective project scope statement includes-
 - The key project objectives
 - Key deliverables
 - Key milestones
 - High-level requirements
 - Assumptions and Exclusions
 - Any known issues or risk
 - Stakeholder review and approval

5. Scope Baseline

- Scope baseline is referred to as the approved version of a scope statement, work breakdown structure (WBS), and its associated WBS dictionary, that can be changed only through formal change control procedures and is used as a basis for comparison.
- A project management plan is a document that describes how the project will be executed, monitored, and controlled.
- It integrates and consolidates all of the subsidiary plans and baselines from the planning processes including scope baseline, along with schedule baseline and cost baseline.

6. Schedule and Cost Baseline

- Schedule baseline is the approved version of a schedule model that can be changed only through formal change control procedure and is used as a basis for comparison to actual results. The project management contains the schedule baseline, used as a reference to compare with the actual results to determine if a change, corrective action, or preventive action is necessary.
- Cost baseline is essentially an important facet of the project management plan that companies use to ensure success. Some of these cost baselines include resource baselines as well as production variations. These measurements of various project performance aspects ensure that cost is evaluated in regards to the overall yield of a particular project.

7. Baseline Management Plans

- A baseline management plan, also called a baseline plan, is a plan for adjusting the various project baselines.
- A baseline in project management is a clearly defined starting point for your project plan.
- It is a fixed reference point to measure and compare your project's progress against.
- This allows you to assess the performance of your project over time.

8. Staffing Management Plans

- Staffing management plan and resource management plans are an important part of project resource management.
- Every project will require resources for executing project activities; the need for both manpower and physical resources
- It is important to select the right staff with the right skills at the right time
- A staffing management plan contains a plan for addressing all aspects of manpower
- It is the staff who will actually complete the project work
- Staff will also consume the majority of the project cost

9. Project Quality and Risks Analysis

- **Project Quality Management** is the process for ensuring that all project activities necessary to design, plan and implement a project are effective and efficient with respect to the purpose of the objective and its performance.
- Risk analysis is the process that figures out how likely that risk will arise in a project. It studies uncertainty and how it would impact the project in terms of schedule, quality and

costs if, in fact, it was to show up. Two ways to analyze risk is quantitative and qualitative.

10. Communication and Documentation

- Facilitate effective project meetings, with priority placed on communicating effectively and in a timely manner.
- Prepare project presentations and reports, giving special attention to the audience's interests.
- Prepare a project communication plan that defines how documents will be distributed, how often, and who will do so.
- Identify collaborative tools to facilitate communication among team members, especially when time and travel constraints might affect a project.

IDENTIFYING JOB ROLES AND RESPONSIBILITIES

- Subject Matter Experts: Amish Singh, Zoya Ahmad
- Product owner: Amish Singh
- Project Manager: Zoya Ahmad
- Technical Lead: Amish Singh
- Software Developer: Rohan Swain
- Software Tester: Nitin Singh, Rohan Swain
- User Acceptance Tester: Rohan Swain, Nitin Singh

PROJECT DEADLINES

S. No.	Task	Date
1	Problem Statements	24-30 th December, 2019
2	Stakeholders & Process Models	30 th December-6 th January, 2020
3	Identifying the Requirements from Problem Statements	6-13 th January, 2020
4	Project Plan	13-20 th January, 2020
5	Project Effort Based on Resources	20-27 th January, 2020
6	Estimation of Project Metrics	27 th January-3 rd February, 2020
7	Modeling UML Use Case Diagrams and Capturing Use Case Scenarios	3 ⁻ 10 th February, 2020
8	E-R Modeling from the Problem Statements	10-17 th February, 2020
9	Identifying Domain Classes from the Problem Statements	17-24 th February, 2020
10	Statechart and Activity Modeling	24 th February-2 nd March, 2020
11	Modeling UML Class Diagrams and Sequence diagrams	2-9 th March, 2020
12	Modeling Data Flow Diagrams	9-16 th March, 2020
13	Implementation	16-23 rd March, 2020
14	Estimation of Test Coverage Metrics and Structural Complexity	23-30 th March, 2020
15	Designing Test Suites	30 th March-7 th April, 2020
16	Deployment Report	7-14 th April, 2020

PROJECT COST



Cost estimation



DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038)
	Zoya Ahmad (RA1811027020028)
	Rohan Swain (RA1811027020051)
	Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor

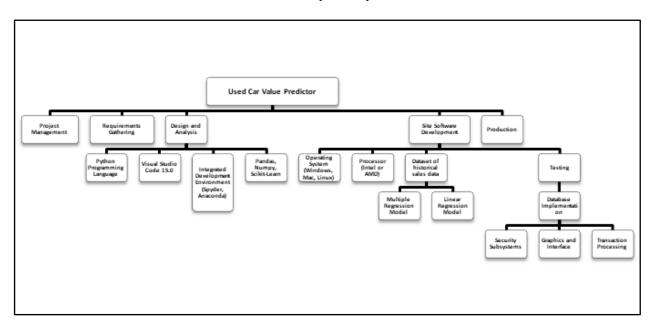


PROJECT EFFORT BASED ON RESOURCES

WORK BREAKDOWN STRUCTURE (WBS)

- The project proceeds with the exploration and collection of data
- The model selection is done through machine learning; linear and multiple regression is used
- We need to check whether the model is predicting accurately or not
- Before listing the car, we need to check whether the car was involved in an accident or criminal activity
- The details about the past owners of the car should be taken
- The details of the car should be entered such as the brand, model, make, no. of kilometres run, mileage, etc.
- The model needs to be trained and tested
- The working model takes around 50 hours to be built

WORK BREAKDOWN STRUCTURE (WBS) CHART



SIZE ESTIMATION TECHNIQUES

Estimation of the size of the software is an essential part of the project. It helps the project manager to further predict the effort and time which will be needed to build the project. Various measures are used in project size estimation. Some of these are—

1. Lines of Code

- Lines of code or LOC is the most popular and used metrics to estimate the size.
- ❖ LOC determination is simple as well. LOC measures the project size in terms of the number of lines of statements or instructions written in the source code. In this count, comments and headers are ignored.
- The size is estimated by comparing it with the existing systems of the same kind.
- ❖ The experts use it to predict the required size of various components of software and then add them to get the total size.
- The size is estimated by comparing it with the existing systems of the same kind. The experts use it to predict the required size of various components of software and then add them to get the total size

2. Number of entities in the ER diagram

- ER model provides a static view of the project.
- It describes the entities and their relationships.
- The no. of entities in the ER model can be used to measure the estimation of the size of the project
- This is because more entitles need more classes/structures, thus leading to more coding
- Size estimation can be done during the initial stages of planning

3. Total number of processes in a detailed data flow diagram

- Data Flow Diagram (DFD) represents the functional view of a software
- The model depicts the main processes/functions involved in software and flow of data between them
- Utilization of the number of functions in DFD to predict software size.
- Already existing processes of similar type are studied and used to estimate the size of the process
- ❖ The sum of the estimated size of each process gives the final estimated size
- It is independent of any programming languages

4. Function Point Metrics

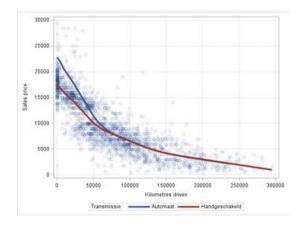
- Function point metrics overcomes many of the shortcomings of LOC.
- Function point metrics proposes that the size of the software project is directly dependent on various functionalities it supports; more the features supported the more would be the size.

❖ This technique helps determine the size of the project directly from the problem specification so is really helpful to project managers during project planning while determining size.

DEPENDENCY GRAPH

- A program is a collection of statements, the ordering and scheduling of which depends on dependence constraints. Dependencies are broadly classified into two categories —
 - Data Dependencies when statements compute data that are used by other statements.
 - 2. **Control Dependencies –** are those which arise from the ordered flow of control in a program.
- A dependency graph can be constructed by drawing edges connect dependent operations.
- These arcs impose a partial ordering among operations that prohibit a fully concurrent execution of a program.
- The use-definition chaining is a form of dependency analysis but it leads to overly conservative estimates of data dependence.
- Four kinds of dependence may exist between statement number 'i' and 'j' on a common control path
 - 1. **Flow Dependence:** Sj is flow-dependent on Si if a value of a variable used by Sj was computed by Si
 - 2. **Anti-dependence:** Sj is anti-dependent on Si if a value of a variable used by Si is recomputed by Sj.
 - 3. **Output-Dependence:** Sj is output-dependent on Si if both compute the same variable and value of the variable from Sj has to be stored after that from Si
 - 4. **Control-Dependence:** Sj is control-dependent on a conditional statement Si if its execution of Si and the path chosen after that (Si must execute before Sj).

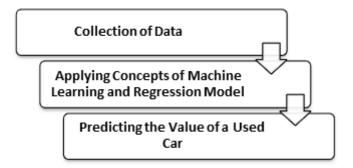
Example of dependency graph is given below -



NETWORK DIAGRAM

- A Network Diagram is a visual representation of a project's schedule.
- Well-known complements to network diagrams include the PERT and Gantt charts.
- A network diagram in project management is useful for planning and tracking the project from beginning to finish.
- It represents a project's critical path as well as the scope for the project.
- There are two types of network diagrams
 - 1. Arrow Diagram: depicts nodes for events and arrows for activities.
 - 2. **Precedence Diagram:** depicts activities in the order they occur.

Example of a network diagram is given below -



TIMELINE

Schedule Inputs:

Several types of inputs are needed to create a project schedule -

- 1. **Personal and project calendars –** Understanding working days, shifts, and resource availability is critical to completing a project schedule.
- Description of project scope From this, you can determine the key start and end dates, major assumptions behind the plan, and key constraints and restrictions. You can also include stakeholder expectations, which will often determine project milestones.
- 3. **Project risks** You need to understand these to make sure there's enough extra time to deal with identified risks and with unidentified risks (risks are identified with thorough Risk Analysis).
- 4. **Lists of activities and resource requirements –** Again, it's important to determine if there are other constraints to consider when developing the schedule. Understanding the resource capabilities and experience you have available as well as company holidays and staff vacations will affect the schedule.

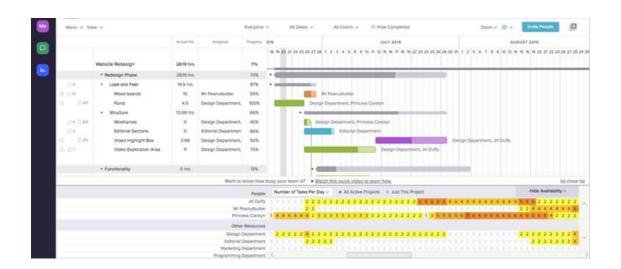
A project manager should be aware of deadlines and resource availability issues that may make the schedule less flexible.

Scheduling Tools:

Some of the tools and techniques for combining these inputs to develop the schedule are -

- Schedule Network Analysis It is a graphic representation of the project's
 activities, the time it takes to complete them, and the sequence in which they must be
 done. Project management software is typically used to create these analyses –
 Gantt charts and PERT Charts are common formats.
- 2. Critical Path Analysis This is the process of looking at all of the activities that must be completed, and calculating the "best line" or critical path to take so that you'll complete the project in the minimum amount of time. The method calculates the earliest and latest possible start and finish times for project activities, and it estimates the dependencies among them to create a schedule of critical activities and dates.
- 3. **Schedule Compression** This tool helps shorten the total duration of a project by decreasing the time allotted for certain activities. It's done so that you can meet time constraints, and still keep the original scope of the project. You can use two methods here:
 - Crashing This is where you assign more resources to an activity, thus decreasing the time it takes to complete it. This is based on the assumption that the time you save will offset the added resource costs.
 - ❖ Fast-Tracking This involves rearranging activities to allow more parallel work. This means that things you would normally do one after another are now done at the same time. However, do bear in mind that this approach increases the risk that you'll miss things, or fail to address changes.

Example of a Gantt chart for Used Car Value Prediction Model -



RISK ANALYSIS

Like any other business that ever took off or the ones just starting, eventually, reach down to this very crucial stage and spend some crucial time deliberating and analysing, questioning their strategy and efforts, with the sole purpose of identifying those instances, or loopholes which if went unnoticed may lead to jeopardizing the assets and profitability of the business. Analysing the risk factors come as a part of the sweet package that includes the idea of coming up with the solution for the same, be it immediate or potential jeopardizing scenarios of the future.

Similarly, the feasibility and the prospective profitability of this Used Car Value predictor project, greatly relies on the number of crucial factors along with the other certain number of risking factors.

This is how the Risk Analysis for the Used Car Value Predictor has been adopted:

DEFINING THE RISKS

- Major jeopardizing factors mainly associated with data and its quality
- Quality is what defines the performance of the model
- Fraudulent instances are also the risks that may deceive the buyers into buys something which were not aware of before.
- Failure in the identification of some vehicle with a criminal/faulty background

SCALING THE RISK AND ITS IMPACTS

- Data of all the prominent brands in the automobile industry market is an essential prerequisite for the functioning, and hence its availability or unavailability puts plenty of affairs on stake.
- Failure in analyzing the data, cleansing of the data and often inappropriate feature scaling can lead to devastatingly incorrect price suggestions.
- Instances of fraudulent exploitations can lead to compromising the new age idea/methodology of adoption of this new reasoned system of price recommendation.

SELECTION AND INCORPORATION OF THE RISK IDENTIFICATION TOOLS/TECHNIQUES

The essentially needed modes for analysing and identifying the risk and other jeopardizing factors for the stated project calls for:

- Documentation Reviews: The standard practice to identify risks is reviewing project-related documents such as lessons learned, articles, organizational process assets, etc.
- Information Gathering Techniques: The given techniques are similar to the techniques used to collect requirements.
- Brainstorming: Brainstorming is done with a group of people who focus on the identification of risk for the project.

- Root Cause Analysis: Root causes are determined for the identified risks. These root causes are further used to identify additional risks.
- Swot Analysis (Strength, Weakness, Opportunities and Threats): Strengths and weaknesses are identified for the project and thus, risks are determined.
- Assumption Analysis: Identification of different assumptions of the project and determining their validity, further helps in identifying risks for the project.
- Risk Data Quality Assessment: Data is collated for the identified risks. The project manager will try to find the precision of the data that must be analyzed for completing the qualitative analysis of risks.
- For each risk, in Risk Data Quality Assessment, the project manager needs to determine:
 - > The extent of the understanding of the risk
 - Data available
 - Quality and reliability of the data
 - > The integrity of the data

DOCUMENTATION OF RISK

- All the impactful and highly important analysis is needed to be laid down on a piece of a legal document, that would clearly define the work and effort flow for the system sustainability.
- This documentation needs to be structured under the above stated risking factors data, quality, instances of fraud etc.
- Structuring must be done in a prioritizing fashion

DOCUMENTATION OF RISK IDENTIFICATION PROCESS

With the application of all the needed risk accessing techniques- Risk data quality assessment, SWOT analysis, Root cause analysis etc; it is essential to bring down the assessment flow for the same onto a legal document for its maintenance in a reviewable format.

ASSESSING THE EFFECTIVENESS

Having documentation of all the accessed risks, a methodology of assessment and prospective solutions can make it easier to test the devised problem/risk tackling plan for its effectiveness through scenario planning, or in the case of actually sharing the front with the accessed problem/risk in a real business transaction

SWOT ANALYSIS

STRENGTH

- A revolutionizing concept of improving the used car industry on reasoned grounds
- Power to leverage the ample amount of past data
- Interface to provide/ quote the price with practically no hassles
- User friendly interface, in an easy to comprehend setting
- Multi-platform support

WEAKNESS

- The complete basis of the project relies on the data and its availability
- Inability to stand up to the stated claims, if a certain case lacks a complete history of available data.

SWOT

OPPORTUNITIES

- Prospective expansion to other foreign land markets
- Collaboration with the automobile giants, contributing as a major asset in their in-house used car business model
- Collaboration with online cab service and rental providers

THREATS

- New entrants with a similar business model
- Obstinate local dealer in the market
- Fraudulent activities
- Instances of unstated claims of some illegal involvements of the concerned vehicle

RISK MITIGATION

RESPONSE	STRATEGY	<u>EXAMPLES</u>
Avoid	 Monitoring the data which feeds, trains and tests the model. Stating the potential fraudulent instances; along with a mechanical validation/verification of the actual status of the vehicle as a part of the advisory. 	 Web scraping for data Stating out all the fraudulent instances faced by most dealers in market
Transfer	 Clearly stating out the accessed policies and risks, especially concerning the data and model usage, for the transference to the third party 	 Proper statement for the transfer of vehicle to take place, in view of the RTO norms.
Mitigate	 One important strategy for the mitigation of all the major potential threats, includes the prior through study and analysis of the current market scenario of the automobile industry; considering all possible parameters- company, brands, type, model, usage etc. 	 Exploration of all the data sources and data-points of the automobile market and its careful analysis to assess the impactful parameters.
Accept	 Putting down all the achievable end- points and non-achievable dead- ends, during the course of prior dealings and deliberation of the budding business/project 	 Scenario planning for tackling situations in a planned fashion

DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038)
	Zoya Ahmad (RA1811027020028)
	Rohan Swain (RA1811027020051)
	Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



ESTIMATION OF PROJECT METRICS

FUNCTION POINT ANALYSIS—

Type of Component	Complexity of Component			
	Low	Average	High	Total
External Inputs	x 3 =	x 4 =	x 6 =	
External Outputs	x 4 =	x 5 =	x 7 =	
External Inquires	x 3 = _	x 4 =	x 6 = _	
Internal Logical Files	x 7=	x 10 = _	x 15=_	
External Interface Files	x 5 = _	x 7 = _	_x 10=_	
Total Number of Unadjusted Function Points				

COCOMO Model—

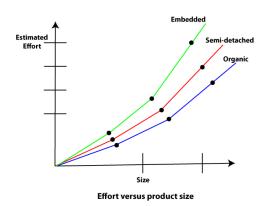
Cocomo (Constructive Cost Model) is a regression model based on LOC, i.e. **number of Lines of Code**. It is a procedural cost estimate model for software projects and often used as a process of reliably predicting the various parameters associated with making a project such as size, effort, cost, time and quality.

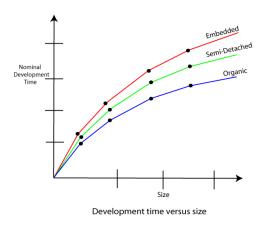
The key parameters which define the quality of any software products, which are also an outcome of the Cocomo are—

- **Effort:** Amount of labour that will be required to complete a task. It is measured in person-months units.
- Schedule: Simply means the amount of time required for the completion of the job, which is, of course, proportional to the effort put. It is measured in the units of time such as weeks, months.

It is of 3 types—

- 1. Embedded
- 2. Semi-Detached
- 3. Organic





Basic COCOMO Equations—

 $E=a_b(kLOC)^b_b$ $D=c_bE^d_b$ P=E/D

Where,

- E is effort in person-months
- D is development time in months
- kLOC is estimated number of lines of code

Software Project	a _b	b _b	C _b	d _b
Embedded	3.6	1.20	2.5	0.32
Semi-	3.0	1.12	2.5	0.35
Detached				
Organic	2.4	1.05	2.5	0.38

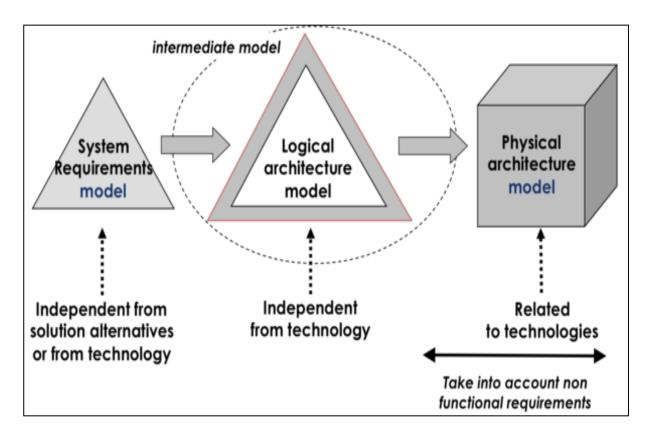
DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038)
	Zoya Ahmad (RA1811027020028)
	Rohan Swain (RA1811027020051)
	Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



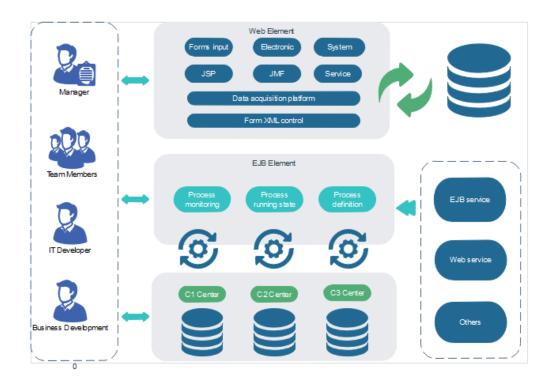
SYSTEM ARCHITECTURE DESIGN

The architecture of a system describes its major components, their relationships (structures), and how they interact with each other. Software architecture and design includes several contributory factors such as Business strategy, quality attributes, human dynamics, design, and IT environment.

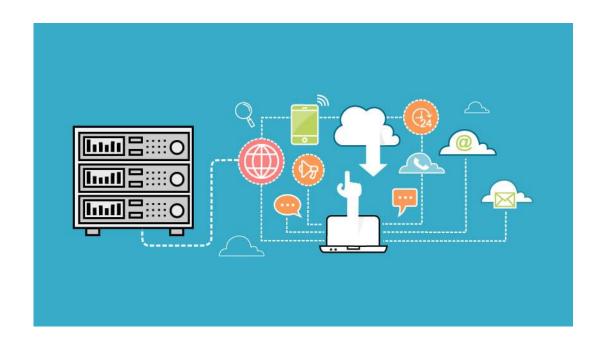
Architecture serves as a **blueprint for a system**. It provides an abstraction to manage the system complexity and establish a communication and coordination mechanism among components. It defines a **structured solution** to meet all the technical and operational requirements, while optimizing the common quality attributes like performance and security.



HIGH LEVEL DIAGRAM—



CLOUD ARCHITECTURE DIAGRAM—



DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038) Zoya Ahmad (RA1811027020028) Rohan Swain (RA1811027020051) Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



MODELLING UML USE CASE DIGRAM & CAPTURING USE CASE SCENARIOS

USE CASE DIAGRAM—

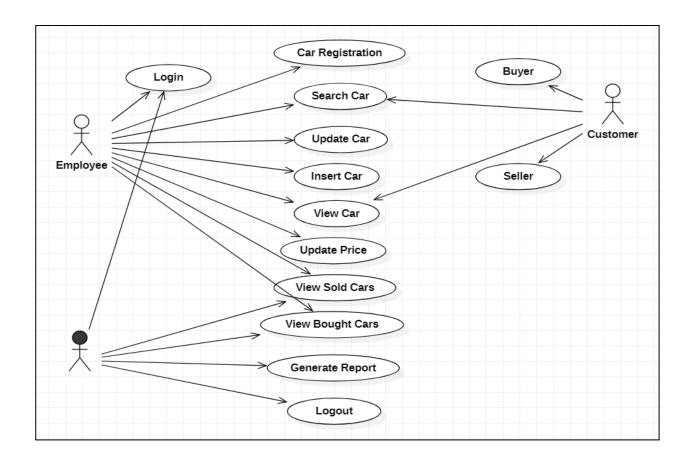
A UML use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behaviour (what), and not the exact method of making it happen (how). Use cases once specified can be denoted both textual and visual representation (i.e. use case diagram). A key concept of use case modelling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behaviour in the user's terms by specifying all externally visible system behaviour.

A use case diagram is usually simple. It does not show the detail of the use cases: It only summarizes some of the relationships between use cases, actors, and systems. It does not show the order in which steps are performed to achieve the goals of each use case. Use case diagrams are considered for high level requirement analysis of a system. When the requirements of a system are analyzed, the functionalities are captured in use cases.

We can say that use cases are nothing but the system functionalities written in an organized manner. The second thing which is relevant to use cases are the actors. Actors can be defined as something that interacts with the system. Actors can be a human user, some internal applications, or may be some external applications.

When we are planning to draw a use case diagram, we should have the following items identified.

- Functionalities to be represented as use case
- Actors
- Relationships among the use cases and actors.



IDENTIFYING ACTORS-

- Employee
- Customer
 - → Buyer
 - → Seller

ASSOCIATION BETWEEN ACTORS AND USE CASE-

- The **employer** hires a System Administrator, a Network Administration, a Network Designer and Technical Team.
- The customer can either be a buyer or a seller.
- The seller registers his car and updates the details of it.
- The buyer searches for the kind of car they want after filling in the requirements.

USE CASE RELATIONSHIPS-

- Include Relationship
- Extend Relationship
- Generalization Relationship

DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038)
	Zoya Ahmad (RA1811027020028)
	Rohan Swain (RA1811027020051)
	Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



E-R MODELLING FROM THE PROBLEM STATEMENTS

ENTITY RELATIONSHIP MODEL—

An **entity–relationship model** (or **ER model**) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between entities (instances of those entity types).

In software engineering, an ER model is commonly formed to represent things a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model that defines a data or information structure which can be implemented in a database, typically a relational database.

Entity–relationship modelling was developed for database and design by Peter Chen and published in a 1976 paper. [1] However, variants of the idea existed previously. Some ER models show super and subtype entities connected by generalization-specialization relationships, and an ER model can be used also in the specification of domain-specific ontologies.

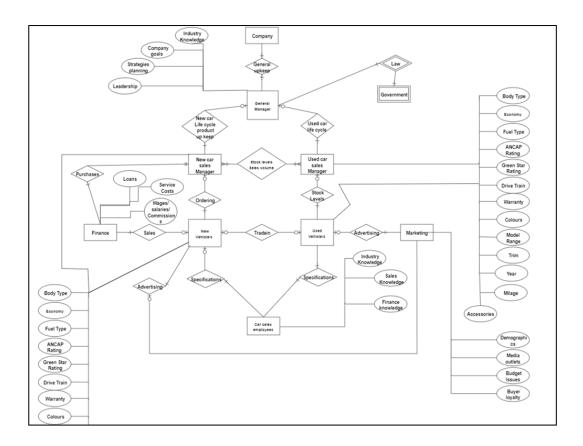
ENTITY SET AND RELATIONSHIP SET—

1. Entity Set

- System Administrator
- Network Administrator
- Network Designer
- Technical Team
- User Actions
- Network Tools
- Other users

2. Relationship Set

- **System Administrator**—System administrator access the feedback section and assign roles to technical team. The software upgrade is done by the system administration only.
- Network Administrator—Network administrator access the network area tools and monitor the network. The
- Other roles under network category are assigned by network administrator.
- Network Designer—Network designer access the analysing tools of network class. The alteration in the network design is done by the network designer.
- **Technical Team**—Technical team upgrade and degrade the software on the basis of feedback and report sanctioned by the system administrator.



2. Attributes of Entity

- **Keys**—User ID will be used as the primary key.
- Candidate Key—User ID , Name , Email ID
- Weak Entity—Network tool is available only when network is available.

DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038) Zoya Ahmad (RA1811027020028) Rohan Swain (RA1811027020051) Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor

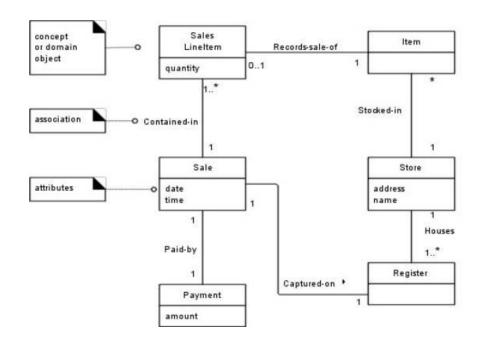


IDENTIFYING DOMAIN CLASSES FROM THE PROBLEM STATEMENTS

A **domain model** is a representation of real-world conceptual classes, not of software components. A **domain model** contains conceptual classes, associations between conceptual classes, and attributes of a conceptual **class**. Informally, a conceptual **class** is an idea, thing, or object.

A domain is a package of related concepts, roles, procedures, events, and entities. Examples of domains include—

- Business
- Engineering
- Science
- Government
- health care



DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038)
	Zoya Ahmad (RA1811027020028)
	Rohan Swain (RA1811027020051)
	Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



STATECHART & ACTIVITY MODELLING

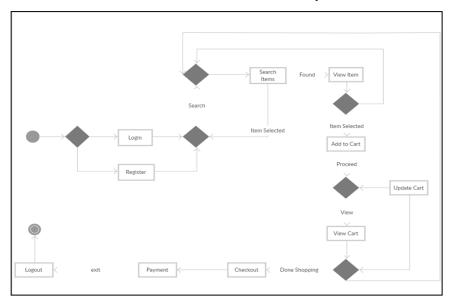
STATECHART DIAGRAMS—

Statechart diagram is one of the five UML diagrams used to model the dynamic nature of a system. They define different states of an object during its lifetime and these states are changed by events. Statechart diagrams are useful to model the reactive systems. Reactive systems can be defined as a system that responds to external or internal events.

Statechart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and it changes when some event is triggered. The most important purpose of Statechart diagram is to model lifetime of an object from creation to termination. Statechart diagrams are also used for forward and reverse engineering of a system. However, the main purpose is to model the reactive system.

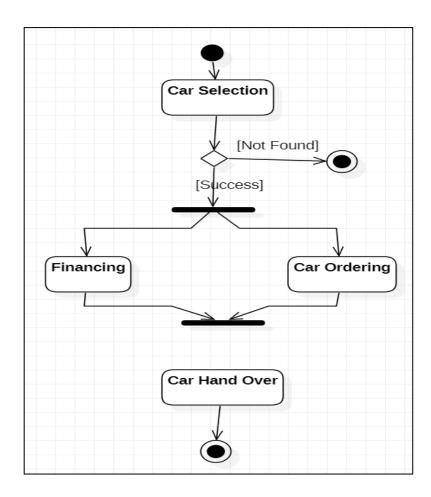
Following are the main purposes of using Statechart diagrams—

- To model the dynamic aspect of a system.
- To model the life time of a reactive system.
- To describe different states of an object during its life time.
- Define a state machine to model the states of an object.



ACTIVITY DIAGRAM—

Activity diagram is defined as a UML diagram that focuses on the execution and flow of the behaviour of a system instead of implementation. It is also called **object-oriented flowchart**. Activity diagrams consist of activities that are made up of actions which apply to behavioural modelling technology.



Components of Activity Diagram—

1. Activities

It is a behaviour that is divided into one or more actions. Activities are a network of nodes connected by edges. There can be action nodes, control nodes, or object nodes. Action nodes represent some action. Control nodes represent the control flow of an activity. Object nodes are used to describe objects used inside an activity. Edges are used to show a path or a flow of execution. Activities start at an initial node and terminate at a final node.

2. Activity partition/Swimlane

An activity partition or a swimlane is a high-level grouping of a set of related actions. A single partition can refer to many things, such as classes, use cases, components, or interfaces.

If a partition cannot be shown clearly, then the name of a partition is written on top of the name of an activity.

3. Fork and Join nodes

Using a fork and join nodes, concurrent flows within an activity can be generated. A fork node has one incoming edge and numerous outgoing edges. It is similar to one too many decision parameters. When data arrives at an incoming edge, it is duplicated and split across numerous outgoing edges simultaneously. A single incoming flow is divided into multiple parallel flows.

A join node is opposite of a fork node as It has many incoming edges and a single outgoing edge. It performs logical AND operation on all the incoming edges. This helps you to synchronize the input flow across a single output edge.

4. Pins

An activity diagram that has a lot of flows gets very complicated and messy. Pins are used to clearing up the things. It provides a way to manage the execution flow of activity by sorting all the flows and cleaning up messy thins. It is an object node that represents one input to or an output from an action. Both input and output pins have precisely one edge.

DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038)
	Zoya Ahmad (RA1811027020028)
	Rohan Swain (RA1811027020051)
	Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



MODELLING UML CLASS DIAGRAM & SEQUENCE DIAGRAM

CLASS DIAGRAM—

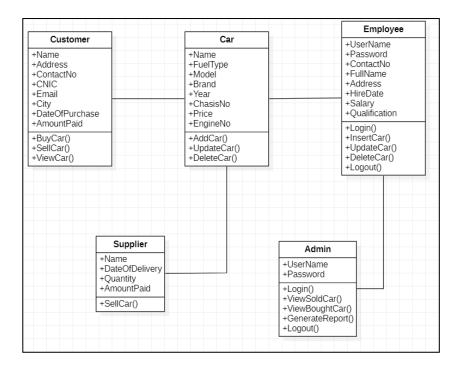
In software engineering, a **class diagram** in the Unified Modeling Language (UML) is **a type of static structure diagram** that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of objectoriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

Class diagram is used to—

- Analysis and design of the static view of an application.
- Describe responsibilities of a system.
- Base for component and deployment diagrams.
- · Forward and reverse engineering.



CLASS-

- User Registration This class is for the registration of the user.
- Generate OTP This class is responsible for the generation of the OTP.
- Login System—this class has all the functionalities related to login.
- Account This class contains all the details related to user account and functions needed to modify and access details.
- **Dashboard** This class contains different modules and sub-routines based on the user type. It also provides access to the account settings.
- History This class contains all history and records of actions performed by every user.
- System Administrator, Network Admin, Network Designer and Technical Team These classes contain all the functionalities related to their particular domain.
- Network Tools This class contains all the network tools like analyse, modify and so on.

RELATIONSHIP—

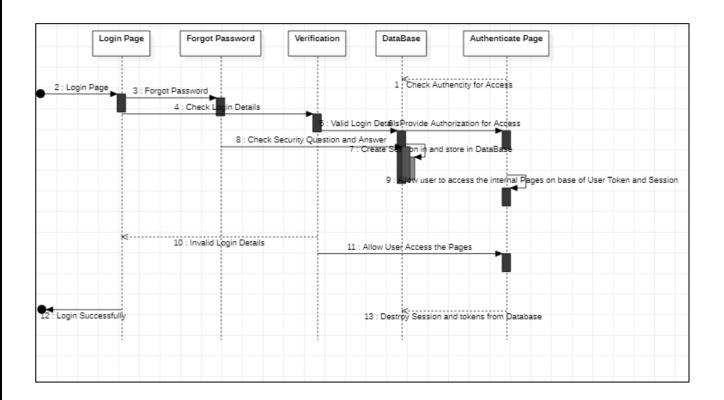
Different classes have different types of relationship with other; some of them have aggregation while many have association between them. For example, User Registration class has composition relationship with OTP Verification class, User and account class have multiplicity relationship, Dashboard and show module class have association relationship between them.

SEQUENCE DIAGRAM—

Unified Modelling Language (UML) is a modeling language in the field of software engineering which aims to set standard ways to visualize the design of a system. UML guides the creation of multiple types of diagrams such as interaction, structure and behaviour diagrams.

A **sequence diagram** is the most commonly used **interaction** diagram. An interaction diagram is used to show the **interactive behavior** of a system. Since visualizing the interactions in a system can be a cumbersome task, we use different types of interaction diagrams to capture various features and aspects of interaction in a system.

A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for new and existing systems.



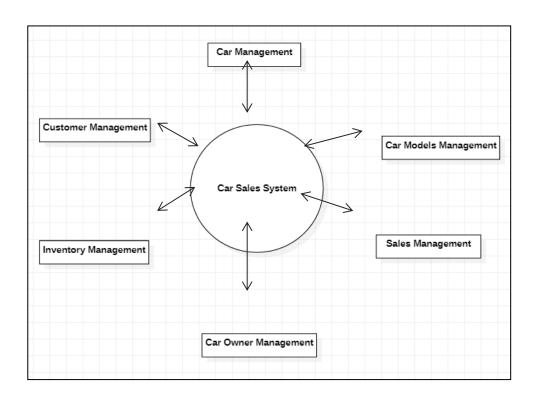
DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038)
	Zoya Ahmad (RA1811027020028)
	Rohan Swain (RA1811027020051)
	Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



MODELLING DATA FLOW DIAGRAM

Also known as DFD, Data flow diagrams are used to graphically represent the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage and reports generation. Data flow diagrams can be divided into logical and physical. The logical data flow diagram describes flow of data through a system to perform certain functionality of a business. The physical data flow diagram describes the implementation of the logical data flow.

DFD graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system. The visual representation makes it a good communication tool between User and System designer. Structure of DFD allows starting from a broad overview and expands it to a hierarchy of detailed diagrams.



DFD has often been used due to the following reasons—

- Logical information flow of the system
- Determination of physical system construction requirements
- Simplicity of notation
- Establishment of manual and automated systems requirements

DFD Symbols—

There are four basic symbols that are used to represent a data-flow diagram—

- Process
- Notation
- Data Flow
- Notation

DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038) Zoya Ahmad (RA1811027020028) Rohan Swain (RA1811027020051) Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



IMPLEMENTATION

ADDITION OF NEW REQUIREMENTS—

New functionalities can be added to the existing project to enhance the user experience and offer a wider domain of functionalities to the user base and potentially to expand it.

Some of the potentially new requirements can be -

- Including the car brands new in the country
- Updating the database with new car models with the launch by the existing brands in the market
- Custom future price prediction functionality for users to estimate the price of their cars in (say) x years from now
- Expanding to including more information about the car

ADDITION OF NEW ISSUES—

- Issue while switching from one network to another network
- Issue while generating two way authentication request
- Issue while dumping log file and saving log file in cloud
- Issue with network analyser on bigger networks

DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038)
	Zoya Ahmad (RA1811027020028)
	Rohan Swain (RA1811027020051)
	Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



ESTIMATION OF TEST COVERAGE METRICS AND STRUCTURAL COMPLEXITY

CONTROL FLOW GRAPH

A control flow graph (CFG) is a directed graph where the nodes represent different instructions of a program, and the edges define the sequence of execution of such instructions.

McCABE'S CYCLOMATIC COMPLEXITY

McCabe had applied graph-theoretic analysis to determine the complexity of a program module. Cyclomatic complexity metric, as proposed by McCabe, provides an upper bound for the number of linearly independent paths that could exist through a given program module. Complexity of a module increases as the number of such paths in the module increase. Thus, if Cyclomatic complexity of any program module is 7, there could be up to seven linearly independent paths in the module. For a complete testing, each of those possible paths should be tested.

OPTIMUM VALUE OF CYCLOMATIC

A set of threshold values for Cyclomatic complexity has been presented below-

V(G)	Module Category	Risk
1-10	Simple	Low
11-20	More complex	Moderate
21-50	Complex	High
> 50	Unstable	Very high

It has been suggested that the Cyclomatic complexity of any module should not exceed 10. Doing so would make a module difficult to understand for humans. If any module is found to have Cyclomatic complexity greater than 10, the module should be considered for redesign.

DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038)
	Zoya Ahmad (RA1811027020028) Rohan Swain (RA1811027020051)
	Nitin Singh (RA1811027020031)
TITLE / ROLE	Used Value Car Predictor



DESIGNING TEST SUITES

SOFTWARE TESTING

The process, to evaluate the functionality of a software application with an intent to find whether the developed software met the specified requirements or not and to identify the defects to ensure that the product is defect-free in order to produce the quality product;

For our software testing involves testing the machine learning model against as many cars and brand as possible and check for their mean squared error and showed be as low as possible. Other checks may involve checking the UI aspects of the same to ensure a smoother use experience

TESTING FRAMEWORKS

Testing frameworks that can be incorporated to ensure the trusted and expected response from the software include-

White Box Testing

White Box Testing is based on application's internal code structure. In white-box testing, an internal perspective of the system, as well as programming skills, is used to design test cases. This testing is usually done at the unit level.

Black Box Testing

Black Box Testing is a software testing method in which testers evaluate the functionality of the software under test without looking at the internal code structure.

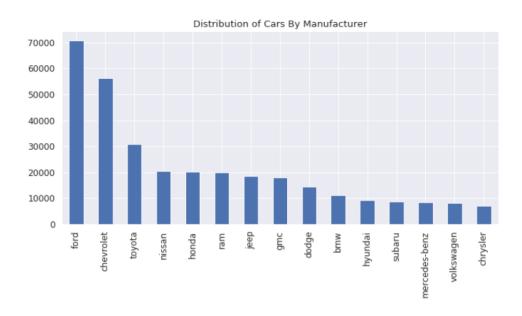
Master Test Plan

The Master Test plan for the given piece of software would to use a automation testing tool such as python's selenium library to construct the code that would automatically test all the UI aspects and also run the main code and log the

result and at any point where it would encounter some error, then it shall reflect the same. Hence it is a very accurate and time optimised approach toward testing the software.

Manual Testing

Manual testing is the process of testing software by hand to learn more about it, to find what is and isn't working. This usually includes verifying all the features specified in requirements documents, but often also includes the testers trying the software with the perspective of their end user's in mind.



DATE	
	Group No. 16
SUBMITTED BY	Amish Singh (RA1811027020038) Zoya Ahmad (RA1811027020028) Rohan Swain (RA1811027020051) Nitin Singh (RA1811027020023)
TITLE / ROLE	Used Value Car Predictor



CONCLUSION

Thus, putting all the aspects of professional approach to formulating a software design plan to its testing into place, a machine learning software is constructed which holds the ability to provide a reliable means to consolidate the existing used car industry by bringing in a reliable and more transparent approach for quoting the price for a car in consideration, given all its parameters that go into deciding the price of the car. The strong statistical basis of the model is where it derives its credibility from.

REFERENCES

- Deeper insights in the used car industry
 - o mordorintelligence.com
 - o cars24.com
- Data Manipulation using pandas www.pandas.pydata.org
- Machine learning implementation using scikit learn www.scikit-learn.org, www.towardsdatascience.com
- Software project management insights www.wrike.com