DSA Cracker

A Project Report

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

By

Pal Pradeepkumar Jilajeet Santra
229702

Under the esteemed guidance of

Dr. Hiren Dand

Coordinator, Department of Information Technology

DEPARTMENT OF INFORMATION TECHNOLOGY
Parle Tilak Vidyalaya Association's

Mulund college of commerce

(Affiliated to university of Mumbai)
NAAC Re-Accredited a Grade – III Cycle
Mulund WEST, MUMBAI 40008
MAHARASHTRA, INDIA 2022-23

PROJECT PROPOSAL

Sr. No.	Description		Details
1	PRN No	:	2020016400098554
2	Roll No	:	229702
3	Name of the Student	:	Pal Pradeepkumar Jilajeet Santra
4	Title of the Project	:	DSA Cracker
5	Name of the Guide	:	Dr. Hiren Dand
6	Teaching Experience of Guide	:	28 years
7	Is this your first Submission	:	Yes

Signature of the Student **Signature of the Coordinator**

Date: …………………… **Date………………………...**

Signature of the Co-ordinate

Date……………………….

Parle Tilak Vidyalaya Association’s

MULUND COLLEGE OF COMMERCE

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled **DSA Cracker** Undertaken

at **PTVA’s MULUND COLLEGE OF COMMERCE** by **Pal Pradeepkumar Jilajeet Santra** Seat no. **229702** in partial fulfilment of B.Sc. IT degree (Semester. IV) Examination has successfully completed the all the phases of the project under my supervision during the academic year 2022 – 23.

Internal Guide and Coordinator

External Examiner

Principal

Date:

ABSTRACT

This document is meant for relating all the procedures that were tracked while developing the DSA Cracker and specially indicates the details of the project and by what means it was developed, the necessities, designs as well as various functions of the project, and the procedures surveyed in achieving these purposes.

Now a time, Data Structure and Algorithm knowledge is necessities to all product-based company that’s why most student need to prepare the question of all the topics in DSA and track them on daily basis and to keep the plan up-to-date.

DSA Cracker is an Web-based application, which can be executed on any Device like Desktop, laptop, Mobile phone (android, iOS). In this application, MongoDB is the Back-end database language and HTML, CSS, JavaScript, React, React-Reveal, Bootstrap is the Front-end language.

As per the daily basis, this application keeps track of your daily question solving in how many time you take to solve your question. This will update the record daily according to calendar.

ACKNOWLEDGEMENT

The existing project, a part of the curriculum, was a first of its- kind experience for me. I had looked upon this project not merely as a syllabus to be completed but as aim to know, study, develop and experience the commercial software enhancement and development strategies.

I would like to acknowledge and appreciate the support of a few, who served a helping hand physically, mentally, and intellectually in the development of this project. The past two years have been so helpful for this project, as software-related planning needs and design, implementation were been explained, thank you subject teachers for letting me understand the need for dedication to project. Foremost regards to my guide, I would thank our **H.O.D, Dr. Hiren Dand** and **Principal, Dr. Sonali Pednekar** who made available the facilities required for the project work such as suitable timings, regular discussion.

I am pleased to be able to say that, I have achieved my goals to make this project a result. I also wish to mention the unsaid support of my parents who, as always helped me in every possible way to make this work of mine, a success, they helped me with a work full environment.

The contribution made by my friends and mates directly or indirectly was essential, and will always be remembered. This opportunity has given me valuable experience in software development in the real world, which will help me in the near future.

Thanking You.

Name

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

“Strings are the favorite data structures of bad programmer.” – James moen

DSA Cracker is a application developed for daily track of your How many question you solve in one day, one week, or one month. It will keep track of all your submission of the question at every time. It will also show how much you reach in the form of level. It will help student to solve question of different type of data structure and various type of algorithm is used to solve the question depend on which type of algorithm best for that problem that will take less time and space to solve the problem.

This application will be categorized by different section of data structure and algorithm to apply different type of approaches to solve the problem to optimize the solution. It will develop the thinking power of the student to think the problem in shortly manner. Data Structure is used to write the efficient and optimize code to speed up the execution of the program.

- What is Data Structure and Algorithm?

  A data structure is a named location that can be used to store and organize data.

  And, an algorithm is a collection of steps to solve a particular problem.

Learning data structures and algorithms allow us to write efficient and optimized computer programs.
Data Structure and algorithm that generally recommended to solve into the 3 language which is C++, Java and Python. C++ and Java both are object-oriented programming languages and high-level programming language. Every beginner Programmer are suggested by to start programming from C/C++ because it will develop your approach to solve the problem and logic building also. It is helpful for the people who want to crack the interview of product-based company like Microsoft, amazon, apple, etc.

1.1 Background

“Bad Programmer worries about the code. Good Programmers worry about data structures and their relationships.” → **Linus Torvalds**

Data Structure are tools used to organize and store data. They can be used for a variety of purposes, including structuring data for storage, retrieving data, and verifying data integrity. There are several different types of data structures, including hierarchical structures, (such as trees and tables), relational structures (such as databases and arrays), network structures (such as graphs), and object-oriented structures (such as classes). Each type has its own advantages and disadvantages, so it is important to choose the right one for the task at hand.

The application of a data structures is numerous ranging from database design to computer programming. Data structures can be used in any Programming language, although their use may vary depending on the language. Data Structures are often used at the beginning of a project will work define how your project will work before you start building it. Data structures can also be used later on in the development process to help you troubleshoot problems with your project as they arise.

- Some of the most important application of data structure include:

i) Data Validation – Checking that a value is of the expected type before using it.

ii) Entropy reduction – making sure that values are distributed uniformly across storage locations.

iii) Redundancy reduction – reducing the total size of a set for space efficiency and security reasons (e.g., disk space)

iv) Structural integrity – ensuring that data is stored in a valid format.

In computer science, the data structures and algorithm is used to organizing the data, managing the data, storing that is usually chosen for efficient access to data. A data structure is a collection of data values, the relationships among them, and the functions or operations that can be applied to the data for example algebraic structure about data.

Different types of data structures are well suited for different kinds of applications, and some are highly specialized to specific tasks. For example, relational databases commonly use B-tree indexes for data retrieval, while complier implementations usually use hash tables to look up identifiers

1.2 Objectives

“It is better to have 100 functions operate on one data structure than to have 10 functions operate on 10 data structures.” → *Alan Peris*

The topics covered will be similar to those found in introductory algorithms and data structures in computer science: sorting and searching algorithms, categorizing efficiency in time and space use, linked list a tree data structures, hash tables, stacks and queues. The objective is that you should know something of all of these by the end of the documentation.

1) If you are the computer science or information technology student then data structure is everything for you.
2) If you take the example of a google, I can simply say that it is completely data structure company. (How efficiently we can search on google it’s because of data structure.)
3) Be familiar with basic techniques of algorithm analysis.
4) Be familiar with writing recursive methods
5) Implementation of data structures such as linked lists and binary trees.

6) Be familiar with advanced data structures such as balanced search trees, hash tables, priority queues an d the disjoint set union/find data structure.
7) Be familiar with several sub quadratic sorting algorithms including quicksort, merge sort, and heapsort.
8) Analyzing problems and writing program solutions to problem using the above techniques.

Data structures are collections of rules that govern how to organize and represent information so that it can be processed efficiently and stored in ways that make it easy to access. These rules include things like how to order records so they can be stored, how to store arrays in memory, and how to map data from one format to another.

The objective of this project:

- To understand the abstract data types stack, queue, deque, and list.
- To be able to implement the ADTs stack, queue and deque using Python list.
- To understand the performance of the implementation of basic linear data structures.
- To understand prefix, infix, and postfix expression formats.
- To use stacks to evaluate postfix expressions.
- To use stack to convert expressions from infix to postfix.
- To use queues for basic timing simulations.
- To be able to recognize problem properties where stacks, queues, and deques are appropriate data structures.
- To be able to implement the abstract data type list as a linked list using the node and reference pattern.

- To be able to compare the performance of our linked list implementation with python’s list.

The main objective of the student is to solve the problem into 3 categories like easy, medium, hard.

If you’re solving easy problem then you need to solve within 10 – 20 min maximum.

If you’re solving medium problem then you need to solve within 30 – 40 min maximum.

If you’re solving hard problem then you need to solve within 1 hours maximum.

1.3 Purpose, Scope & Applicability

1.3.1 Purpose

“Smart data structures and dumb code works a lot better than the other way around..” → Erik’s. Raymond

In our day today life, everything is about data. Yes, we have lots of data to play with, but to do that we need a proper place to store it and use it back. For example, we cannot store water in a bag, we store it in a bottle and we can’t put vegetables or eggs in a bottle, we use a bag. Every container is designed to store specific items (here different data types/formats). So, learning data structure helps us to store the data we have in an efficient manner to retrieve it with low cost and less time. After using the efficient data structure, we can extract the information we want or process it for further analysis.

Let us try and see this using an example.

Suppose we have this list of tasks that is always performed in an organization.

1. add a new employee.
2. given a employee, get his team leader.
3. given a team leader, give a list of people under him.
4. Get employee details (Like date of joining, position, no of people working under him etc.)
5. Check whether an employee works in a certain department

And we want to do all these tasks.

A naive way to this would be for every employee, create a file by the name of employee id, and store all the data inside the file. We will not be able to return the list of all employees under him in constant time, but we managed to do all other stuff. And i didn't even used any data structure so far.

But there is some problem here. Suppose when we promote an employee, what all are the things that changed. Now he may be a team leader or member in some other department. Now we have to change the data of all the associated files. For a large organization, this task is extremely tedious and time consuming.

What could we have done differently?

The answer is simple, try and create a data structure that can handle such task.

Let us try to structure everything in the form of a tree now.

1. adding a employee now just means adding a node with employee data (the basic ones).
2. The team leader is now the parent of the node.
3. the list of the employees working under are now just the child nodes.
4. getting the data of the employee can be done from the data field
5. go to a certain department and just check whether a node is in a given sub tree.

Promoting employee now just means swapping of nodes. Demoting works in similar fashion.

I realize that this is not a complete explanation but i hope it gives you an idea of importance of data structures in handling, maintaining and updating of data.

This is just a tool to make your everyday task easier.

What is the importance of DSA Cracker in a college?

Every college have a data structure in their syllabus to study all form of data structure and algorithm for better understanding all the format of question to organize in order. Data structure is every thing for all computer science and information technology student in the industries.

Data structures and algorithms are used to solve problems and teach developers how to design effective code. The amount of memory and time the code uses can be used to assess its quality. The more quickly the code executes and uses less memory, the more efficient it is. DSA plays a significant role in software businesses' hiring procedures as well. A DSA is used by recruiters to evaluate a candidate's programming skills since it demonstrates their ability to solve problems.

Why are Data structures and algorithms (DSA) important in Computer Science?

The importance of DSA in lowering the code's temporal complexity cannot be overstated. There may be various ways to solve an issue, but in order to stand out from the competition, you must choose the best strategy. For each issue statement given, you must be able to develop code that executes more quickly. It can be accomplished by studying DSA. If you have a thorough understanding of DSA, writing efficient code will come naturally to you**.** You will acquire the knowledge and abilities necessary to make your code less difficult.

Let's delve more into the different benefits of learning DSA:

1. DSA's Function in Real-World Problem Solving
2. DSA's function in machine learning
3. DSA's function in the technical hiring process.
4. Computer science's foundation.

Learning new programming languages while utilizing them in your projects is rather simple. DSA, however, is an exception. It will be difficult to optimize the code and solution for an issue if you are unfamiliar with DSA. Because of DSA's many applications, it is essential for all aspiring programmers and students to begin understanding it in order to advance in the field of computer science. In order to stay current with DSA's improvements, it is also crucial for the individuals who presently work in it to upgrade their skills.

Tracking student time will also let them know if they are punctual. This only works if you require students to document the exact time they enter or leave. Attendance management lets you save track of how many days off scholars use. This is vital if your education system has a policy that lets students use a certain number of sick or vacation days, for example, 75 % attendance is mandatory to appear in Sem exams.

1.3.2 Scope

*"*The programmer's primary weapon in the never-ending battle against slow system is to change the intramodular structure. Our first response should be to reorganize the modules' data structures.*"* → *Fred Brooks*

The data structure is a process through which data is stored and arranged in the disk space of the computer or memory storage, in a way that the data can be easily used and manipulated in the future.

DSA helps the implementation more efficiently in terms of time and space. While programming languages are used for building applications and user-friendly interfaces. Both are equally important to build full-fledged and efficient applications for the users.

For e.g.- Google maps are used by millions of users, a very efficient, user-friendly, and easy-to-use app. Programmers, while creating this app used DSA, distributed servers, architecture, and programming languages.

To be a good programmer you should be-

- Good at DSA
- Good knowledge about database-related concepts
- Good knowledge of networking concepts
- Good knowledge about microservices architectures
- Good at user-friendly interfaces

It is easier to learn technologies than learning data structures and algorithms. If you are good at Linux, it becomes all the easier for you to learn new technologies.

Most of the time, the work you do would be mostly related to technologies and how you use them in your projects.

Now you might wonder why would I want to learn Data Structures and Algorithms if that is the case. Yes, that is a good question to ask.

1.3.3 Applicability

"Data dominates. If you're chosen the right data structures and organized things well, the algorithms will almost always be self-evident. Data structures, not algorithms, are central to programming." → Rob Pike

Types of Data structures:

1) Array
2) String
3) Linked list
4) Stack
5) Queue
6) Tree
7) Graph

Application of these data structure:-

1) **Array**: Arrays are the simplest data structures that store items of the same data type. A basic application of Arrays can be storing data in tabular format. For example, if we wish to store the contacts on our phone, then the software will simply place all our contacts in an array.

- Arrangement of the leader-board of a game can be done simply through arrays to store the score and arrange them in descending order to clearly make out the rank of each player in the game.
- A simple question Paper is an array of numbered questions with each of them assigned some marks.
- 2D arrays, commonly known as, matrices, are used in image processing.
- It is also used in speech processing, in which each speech signal is an array.
- Your viewing screen is also a multidimensional array of pixels.

- Book titles in a Library Management Systems.
- Online ticket booking.
- Contacts on a cell phone.
- For CPU scheduling in computer.
- To store the possible moves of chess on a chessboard.
- To store images of a specific size on an android or laptop.

2) String:

- Spam email detection.
- Plagiarism detection.
- Search engine.
- Digital forensic and information retrieval system
- Spell checkers.
- In the database to check valid information of the user

3) **Linked List:** A Linked List is a sequence data structure, which connects elements, called nodes, through links.

Some other applications of the linked list are:

- Images are linked with each other. So, an image viewer software uses a linked list to view the previous and the next images using the previous and next buttons.
- Web pages can be accessed using the previous and the next URL links which are linked using a linked list.
- The music players also use the same technique to switch between music.
- To keep the track of turns in a multi-player game, a circular linked list is used.
- MS-Paint drawings and shapes are connected via a linked list on canvas.
- Escalators — Circular linked List.
- Each of the lines of code in an IDE internally is a record on a doubly-linked list.
- Left/Right swipe on Tinder uses a doubly-linked list.

- Social media content “feeds”.
- Used for symbol table management in a designing compiler

4) **Stacks**:- A stack is a data structure that uses LIFO Order.

- Converting infix to postfix expressions.
- Undo/Redo button/operation in word processors.
- Syntaxes in languages are parsed using stacks.
- It is used in many virtual machines like JVM.
- Forward-backward surfing in the browser.
- History of visited websites.
- Message logs and all messages you get are arranged in a stack.
- Call logs, E-mails, Google photos’ any gallery, YouTube downloads, Notifications (latest appears first).
- Scratch card’s earned after Google pay transaction.
- Wearing/Removing Bangles, Pile of Dinner Plates, Stacked chairs.
- Changing wearables on a cold evening, first in, comes out at last.

5) **Queue**:- A queue is a data structure that uses FIFO order.

- Operating system uses queues for job scheduling.
- To handle congestion in the networking queue can be used.
- Data packets in communication are arranged in queue format.
- Sending an e-mail, it will be queued.
- Server while responding to request
- Uploading and downloading photos, first kept for uploading/downloading will be completed first (Not if there is threading)
- Most internet requests and processes use queue.
- While switching multiple applications, windows use circular queue.
- In Escalators, Printer spooler, Car washes queue.

- A circular queue is used to maintain the playing sequence of multiple players in a game.
- A queue can be implemented in-Linked List-based Queue, Array-based Queue, Stack-based Queue.
- Uploading and downloading photos, first kept for uploading/downloading will be completed first (Not if there is threading).
- Handle website traffic
- CPU scheduling

6) **Tree**:- Trees are hierarchical structures having a single root node.

- XML Parser uses tree algorithms.
- The decision-based algorithm is used in machine learning which works upon the algorithm of the tree.
- Databases also use tree data structures for indexing.
- Domain Name Server(DNS) also uses tree structures.
- File explorer/my computer of mobile/any computer
- BST used in computer Graphics
- Posting questions on websites like Quora, the comments are a child of questions.
- Parsers(XML parser).
- Code Compression(zip).
- DOM in Html.
- Evaluate an expression (i.e., parse).
- Integral to compilers/automata theory.
- To store the possible moves in a chess game.
- To store the genealogy information of biological species.
- Used by JVM (Java Virtual Machine) to store Java objects.

7) **Graph**:- Graph is a data structure where data is stored in a collection of interconnected vertices (nodes) and edges (paths).

- Facebook’s Graph API uses the structure of Graphs.
- Google’s Knowledge Graph also has to do something with Graph.
- Dijkstra algorithm or the shortest path first algorithm also uses graph structure to find the smallest path between the nodes of the graph.
- The GPS navigation system also uses shortest path APIs.
- Networking components have a huge application for graph
- Facebook, Instagram, and all social media networking sites every user is Node
- Data organization
- Reacts virtual DOM uses graph data structures.
- MS Excel uses DAG (Directed Acyclic Graphs).
- Path Optimization Algorithms, BFS, DFS.
- Recommendation Engines.
- Scientific Computations, Flight Networks, Page ranking.
- Google map to find nearest location.
- Facebook to suggest mutual friends

1.4 Achievements

The achievements that were achieved from this project that is including various diagrams, models, explanation through charts.

Various type of idea been developed on the basis of digitalization. Implementation of this project will give us a rough idea about the type of question been asked in the interview by the company through placement cell.

The use of web development technologies like HTML, CSS, JavaScript, React, React-Reveal, MongoDB.etc. Nowadays Frameworks playing an important role in the programming language to make such thing simple and easy to all.

1.5 Organisation of Report

Chapter 1: Introduction

Following are several outlines:

a. Background: The background shows the implementation of the project how the problem will take step to compile all the program.

b. Objectives: Project was set the goal of the user through this.

c. Purpose: Project requirements descriptions that answers questions about the need of implementation the project.

d. Scope: This project will cover all the topic which you are going to face while in the interview.

e. Applicability: The project will improve the student performance to solve the problem with their new ideas and logic.

f. Achievements: Achievement of the student after the project was success.

Chapter 2: Review of Technologies:

The Technology Survey validates the acquaintance and understanding of available technology associated with the project. Describe the technologies obtainable in the chosen area and present a reasonable study of all available technologies and which is the best of them and why choose the technology.

Kotlin language is used for making this application. Kotlin is a cross-platform programming language it is intended to fully interwork with Java but it lets its syntax to be more concise. JVM is the main target of Kotlin. JetBrains and Google sponsor Kotlin language. Google officially supports Kotlin for Android, as Kotlin is an alternative to java.

Chapter 3: Software Requirement Specifications:

This chapter defines what problem you are experiencing and how your project will overcome the problem that is occurring in society. Listing the necessities of the project. Scheduling and planning for the project so that project should complete on time and doesn't go on a critical path. Give the list of software along with hardware components require by the project and explain key points about it. Charts, models, and diagrams

Chapter 4: System Design:

This chapter describes features and consists of designing a system, i.e., designing schemes, designing algorithms, and designing basic modules of a system

Chapter 5: Implementation and Testing

This chapter contains the details of various modules, codes, and various libraries I have imported successfully to implement the project. It also contains the details of the process or work I have done to make code efficient. This chapter also contains the details of various testing approaches I have used.

Chapter 6: Results and Discussion

This Chapter shows the details of the test results after testing the software and generates a report based on the Test results. This chapter will also show the behaviour of the application when inputs are different from the ones written in the Test Cases.

This Chapter also contains details of the working of the application and the different functions in the project. It should also contain the User Manual, which provides the understanding of the working of the project to the user.

Chapter 7: Conclusion

This chapter summarizes all the important points of all other chapters. It also shows the limitation in the proposed system and details about a plan to explore the scope of the project

CHAPTER 2: REVIEW OF TECHNOLOGIES

2.1 Technologies available for development:

“To write a kernel without a data structure and have it been as consistent and graceful as UNIX would have been a much, much harder challenge.” - Bill Gates

In the present time, there are various platform to practice data structure and algorithm and many other types of language coding question also. Platform like, Hackerrank, GeekForGeek, Leetcode, CodeForces, etc. but this platform is not for beginner-friendly because the problem in this platform is very tricky that’s why the coder need to think logically.

There are many categories in this platform like: -

1) Arrays
2) Strings
3) Linked List
4) Stack
5) Hash
6) Sorting and Searching
7) Mathematical question
8) Tree
9) Graph
10) Greedy
11) Binary search tree
12) Algorithm
13) AVL-Tree
14) Machine Learning
15) Dynamic Programming

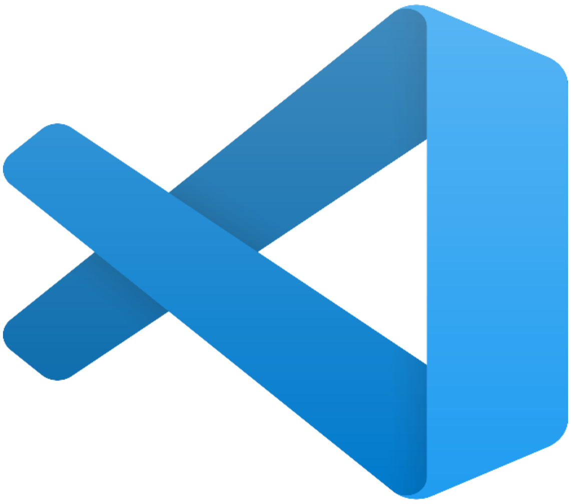
This are the topics of the data structure and algorithm to solve the question in the category wise and you can also pick the question randomly to your choice.

Programming language was used to solve the problem in this platform. Language like C++, Java, Python, C#, C, PHP, JavaScript, Go, etc. But C++ is best and fast language that are suggested to solve the problem.

In C++, there is STL Standard Template Library is a software library originally designed by Alexandra Stepanov for the C++ programming language that influenced many parts of the C++ Standard Library. It provides four components called algorithms, containers, functions, and iterators. STL is very useful to solve data structure problem very easily because there is inbuilt data structure are available like vector, LinkedList, stack, hash. It is used to upgrade the space during runtime of the problem.

2.2 Software and Languages Required:

2.2.1. VS Code:



Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including Java, JavaScript, Go, Node.js, Python, C++, C, Rust and Fortran. It is based on the Electron framework, which is used to develop Node.js web applications that run on the Blink layout engine. Visual Studio Code employs the same editor component (codenamed "Monaco") used in Azure DevOps (formerly called Visual Studio Online and Visual Studio Team Services)A flexible construction machine based primarily on Gradle.

- **Support for multiple programming language:** Supports multiple programming languages. So earlier, programmers needed Web-Support: a different editor for different languages, but it has built-in multi-language support. This also means it easily detects if there's any fault or cross-language reference, it'll be able to detect it easily.

- **Intelli-Sense:** It can detect if any snippet of code is left incomplete. Also, common variable syntaxes and variable declarations are made automatically. Ex: If a certain variable is being used in the program and the user has forgotten to declare, intelli-sense will declare it for the user.
- **Cross-platform support:** Traditionally, editors used to support either Windows or Linux or Mac Systems. But Visual Studio Code is cross-platform. So, it can work on all three platforms. Also, the code works on all three platforms; else, the open-source and proprietary software codes used to be different.
- **Extensions and Support:** Usually supports all the programming languages but, if the user/programmer wants to use the programming language which is not supported then, he can download the extension and use it. And performance-wise, the extension doesn't slow down the editor as it rums as a different process.
- **Repository:** With the ever-increasing demand for the code, secure and timely storage is equally important. It is connected with Git or can be connected with any other repository for pulling or saving the instances.
- **Web-Support:** Comes with built-in support for Web applications. So web applications can be built and supported in VSC.
- **Hierarchy Structure**: The code files are located in files and folders. The required code files also have some files, which may be required for other complex projects. These files can be deleted as per convenience.
- **Improving Code**: Some code snippets can be declared a bit differently, which might help the user in the code. This function prompts the user, wherever necessary, to change it to the suggested option.

- **Terminal Support:** Many of the times, the user needs to start from the root of the directory to start with a particular action, in-built terminal or console provides user support to not to switch in-between two screens for the same.
- **Multi-Projects:** Multiple projects containing multiple files/folders can be opened simultaneously. These projects/folders might or might not be related to each other.
- **Git Support:** Resources can be pulled from Git Hub Repo online and vice-versa; saving can be done too. Resource pulling also means cloning the code which is made available on the internet. This code can later be changed and saved.
- **Commenting:** A common feature, but some of the languages do not support it. Commenting on the code helps the user to recall or track according to the sequence he wants.

Another feature that naïve-users or anyone can see instantly different from other editors is the user-friendliness of the Visual Studio Code. The usability is very easy to handle. The file is arranged hierarchically and has regular software like a toolbar, status bar, and a sidebar. It also has a floating windows explorer window, which can be fixed at one place according to convenience, which consists of the directory structure of files. These files (code files, image folders, etc.) can be opened or renamed from here, and changes will automatically get reflected in the storage.

2.2.2.HTML:

HTML (Hypertext Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation (CSS) or functionality/behavior (JavaScript).

"Hypertext" refers to links that connect web pages to one another, either within a single website or between websites. Links are a fundamental aspect of the Web. By uploading content to the Internet and linking it to pages created by other people, you become an active participant in the World Wide Web.

HTML uses "markup" to annotate text, images, and other content for display in a Web browser. HTML markup includes special "elements" such as <head>, <title>, <body>, <header>, <footer>, <article>, <section>, <p>, <div>, <span>, <img>, <aside>, <audio>, <canvas>, <datalist>, <details>, <embed>, <nav>, <output>, <progress>, <video>, <ul>, <ol>, <li> and many others.

Android code is generated once and for running needs to combine and streamline local code for better execution across different gadgets. Java has phase independent components, so it is used for Android advance. Java is an extremely popular language due to its excellent layout and execution. The group of capable developers is huge. As result, Android engineers choose Java because there are now a large amount of Java developers who can help build and develop Android applications in all to the many Java libraries and devices that make their life easier. designers become easier.

An HTML element is set off from other text in a document by "tags", which consist of the element name surrounded by "<" and ">". The name of an element inside a tag is case insensitive. That is, it can be written in uppercase, lowercase, or a mixture. For example, the <title> tag can be written as <Title>, <TITLE>, or in any other way. However, the convention and recommended practice is to write tags in lowercase.

Every language has its qualities and shortcomings. It's based on software engineers to choose which language best serves for specific company. In addition to countless accessible languages, java written computer programs are clearly the most famous language of engineers. Java attaches great importance to its amazing elements that combine solid, superior memory frames, in the same reversal sense and safety of specific features to provide some examples. The last Java is being used in a large number of areas, including the progress of versatile applications, improving web programs and frames.
However, its function is its ability to create small modules or applets for applications.

1) **Introduction to HTML**: This module sets the stage, getting you used to important concepts and syntax such as looking at applying HTML to text, how to create hyperlinks, and how to use HTML to structure a web page.
2) **Multimedia and embedding:** This module explore how to use HTML to include multimedia in your web pages, including the different ways that images can be included, and how to embed video, audio, and even entire other webpages.

3) **HTML tables:** Representing tabular data on a webpage in an understandable, accessible way can be a challenge. This module covers basic table markup, along with more complex features such as implementing captions and summaries.

4) **Html forms:** Forms are a very important part of the Web — these provide much of the functionality you need for interacting with websites, e.g. registering and logging in, sending feedback, buying products, and more. This module gets you started with creating the client-side/front-end parts of forms.

5) **Use HTML to solve common problems:** Provides links to sections of content explaining how to use HTML to solve very common problems when creating a web page: dealing with titles, adding images or videos, emphasizing content, creating a basic form, etc.

2.2.3.CSS:

Cascading Style Sheets (**CSS**) is a stylesheet language used to describe the presentation of a document written in HTML or XML (including XML dialects such as SVG, MathML or XHTML). CSS describes how elements should be rendered on screen, on paper, in speech, or on other media.

CSS is among the core languages of the **open web** and is standardized across Web browsers according to W3C_specifications. Previously, the development of various parts of CSS specification was done synchronously, which allowed the versioning of the latest recommendations. You might have heard about CSS1, CSS2.1, or even CSS3. There will never be a CSS3 or a CSS4; rather, everything is now CSS .

1) **CSS first steps**: CSS (Cascading Style Sheets) is used to style and layout web pages — for example, to alter the font, color, size, and spacing of your content, split it into multiple columns, or add animations and other decorative features. This module provides a gentle beginning to your path towards CSS mastery with the basics of how it works, what the syntax looks like, and how you can start using it to add styling to HTML.

2) **CSS building blocks**: This module carries on where CSS first steps left off — now you've gained familiarity with the language and its syntax, and got some basic experience with using it, it's time to dive a bit deeper. This module looks at the cascade and inheritance, all the selector types we have available, units, sizing, styling backgrounds and borders, debugging, and lots more.

The aim here is to provide you with a toolkit for writing competent CSS and help you understand all the essential theory, before moving on to more specific disciplines like text styling and CSS layout.

3) **CSS Styling Texts**: With the basics of the CSS language covered, the next CSS topic for you to concentrate on is styling text — one of the most common things you'll do with CSS. Here we look at text styling fundamentals, including setting font, boldness, italics, line and letter spacing, drop shadows, and other text features. We round off the module by looking at applying custom fonts to your page, and styling lists and links.

4) **CSS layout**: At this point we have already looked at CSS fundamentals, how to style text, and how to style and manipulate the boxes that your content sits inside. Now it is time to look at how to place your boxes in the right place in relation to the viewport, and to each other. We have covered the necessary prerequisites so we can now dive deep into CSS layout, looking at different display settings, modern layout tools like flexbox, CSS grid, and positioning, and some of the legacy techniques you might still want to know about.

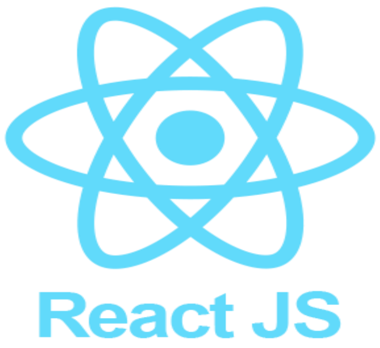
2.2.4. JavaScript:

JavaScript (**JS**) is a lightweight, interpreted, or just-in-time compiled programming language with first-class functions. While it is most well-known as the scripting language for Web pages, many non-browser environments also use it, such as Node.js, Apache CouchDB and Adobe Acrobat. JavaScript is a prototype-based, multi-paradigm, single-threaded, dynamic language, supporting object-oriented, imperative, and declarative (e.g. functional programming) styles. Read more about JavaScript. This section is dedicated to the JavaScript language itself, and not the parts that are specific to Web pages or other host environments. For information about APIs that are specific to Web pages, please see Web APIs and DOM.

1) **JavaScript First Steps**: Answers some fundamental questions such as "what is JavaScript?", "what does it look like?", and "what can it do?", along with discussing key JavaScript features such as variables, strings, numbers, and arrays.
2) **JavaScript Building Blocks**: Continues our coverage of JavaScript's key fundamental features, turning our attention to commonly-encountered types of code blocks such as conditional statements, loops, functions, and events.

3) **Introduction to JavaScript Objects:** The object-oriented nature of JavaScript is important to understand if you want to go further with your knowledge of the language and write more efficient code, therefore we've provided this module to help you.
4) **Asynchronous JavaScript:** Discusses asynchronous JavaScript, why it is important, and how it can be used to effectively handle potential blocking operations such as fetching resources from a server.
5) **Client-side web APIs:** Explores what APIs are, and how to use some of the most common APIs you'll come across often in your development work.

2.2.5: React.JS:



ReactJS tutorial provides basic and advanced concepts of ReactJS. Currently, ReactJS is one of the most popular JavaScript front-end libraries which has a strong foundation and a large community.

ReactJS is a **declarative**, **efficient**, and flexible **JavaScript library** for building reusable UI components. It is an open-source, component-based front end library which is responsible only for the view layer of the application. It was initially developed and maintained by Facebook and later used in its products like WhatsApp & Instagram.

Our ReactJS tutorial includes all the topics which help to learn ReactJS. These are ReactJS Introduction, ReactJS Features, ReactJS Installation, Pros and Cons of ReactJS, ReactJS JSX, ReactJS Components, ReactJS State, ReactJS Props, ReactJS Forms, ReactJS Events, ReactJS Animation and many more.

The main objective of ReactJS is to develop User Interfaces (UI) that improves the speed of the apps. It uses virtual DOM (JavaScript object), which improves the performance of the app. The JavaScript virtual DOM is faster than the regular DOM. We can use ReactJS on the client and server-side as well as with other frameworks. It uses component and data patterns that improve readability and helps to maintain larger apps.

2.2.6 Express.JS:

Express is a minimal and flexible Node.js web application framework that provides a robust set of features to develop web and mobile applications. It facilitates the rapid development of Node based Web applications. Following is some of the core features of Express framework.

- Allows to set up middleware's to respond to HTTP Requests.
- Defines a routing table which is used to perform different actions based on HTTP Method and URL
- Allows to dynamically render HTML Pages based on passing arguments to templates

2.2.6 MongoDB:

mongoDB

MongoDB is an open-source document database and leading NoSQL database. MongoDB is written in C++. This tutorial will give you great understanding on MongoDB concepts needed to create and deploy a highly scalable and performance-oriented database.

MongoDB is a cross-platform, document oriented database that provides, high performance, high availability, and easy scalability. MongoDB works on concept of collection and document.

Collection is a group of MongoDB documents. It is the equivalent of an RDBMS table. A collection exists within a single database. Collections do not enforce a schema. Documents within a collection can have different fields. Typically, all documents in a collection are of similar or related purpose.

A document is a set of key-value pairs. Documents have dynamic schema. Dynamic schema means that documents in the same collection do not need to have the same set of fields or structure, and common fields in a collection's documents may hold different types of data.

Any relational database has a typical schema design that shows number of tables and the relationship between these tables. While in MongoDB, there is no concept of relationship.

2.3 Existing technologies:

“Unlike existing, new technology requires effort.”

2.3.1 Similar software:

2.3.1.1 Visual Studio:

Microsoft Visual Studio is an IDE formed by Microsoft and used for different types of software development such as computer programs, websites, web applications, web services, and mobile applications. It covers finalizers, compilers, and other features to aid the software development process. Visual Studio has been around for over 20 years. Other built-in tools include a compiler, a designer for building GUI applications, a web designer, a class designer, and a database schema designer. Visual Studio IDE has a lot of extensions available in its market. Visual Studio Extensions includes extensions available for C++ (this is the most popular IDE for C++), including Incredibility.

Visual studio features:

- Visual Studio Code is source code editor that is used with many programming languages including Java, JavaScript, Go, Node.js, Python and C++.
- It is depended on Electron framework, used in development of Node.js web applications running on layout engine.
- It supports many different programming languages and feature sets on each. Uninvited files and folders can be excluded from project tree through settings.

- Visual Studio Code allows the user set up the code page in which the active document is saved, the line break character, allowing it be used on different platform, in any location and for any certain programming language.

Visual Studio Code as an editor that values simplicity over a multitude of bells and whistles. Most of the features are displayed by typing what you need in the top search bar. Visual Studio Code can't do multiple projects at once and it looks like it will get this feature soon.

2.3.1.2 Eclipse IDE:

Eclipse is an integrated development environment (IDE) used in computer programming. Eclipse is written primarily in Java and its primary use is Java application development, but it can also be used to develop applications in other programming languages through plugins including Ada , ABAP, C, C++, C#, Clojure, COBOL, D, Erlang .Eclipse uses plugins to provide all the functionality inside and above the runtime system.

The Eclipse SDK consist the Eclipse Java Development Tool (JDT), which contributes an IDE with a built-in Java-incremental-compiler and a complete model. This enables advanced code analysis and refactoring techniques. It supports remote debugging, allowing users to view the variables and step-by-step code of an application running on the connected server. Features of Eclipse IDE:

- The software update system is an important mechanism. For eclipse IDE, software update system is easy as developer can add plugins which help developer to promote continuous release of application.

- Demand is limited in eclipse platform as the IDE provides the capability by providing a simple dialog in which the user selects the required software and plugins in eclipse platform.
- The platform supports JPA/JSF tools, WSD, and supports editing XML files.
- Spring developers also use the same IDE platform as eclipse which also support Spring based applications.
- Testing can done easily using Eclipse IDE. Debugging can run smooth to find bugs in the application.
- Plugins available for Eclipse IDE and users can find plugins in Eclipse Marketplace. Different types of plugins such as static analysis/code style checking, system version checking are available in Eclipse Marketplace.

Android Studio is faster than Eclipse. It is not necessary to add a plugin to Android Studio, but if we are using Eclipse, we need it. Eclipse takes a lot of resources to get started, but Android Studio doesn't.

2.3.1.3 Angular:

Another JavaScript Framework like AngularJS. It is similar to Reactjs but it is slower than reactjs.

AngularJS is a very powerful JavaScript Framework. It is used in Single Page Application (SPA) projects. It extends HTML DOM with additional attributes and makes it more responsive to user actions. AngularJS is open source, completely free, and used by thousands of developers around the world. It is licensed under the Apache license version 2.0.

- AngularJS is a efficient framework that can create Rich Internet Applications (RIA).
    - AngularJS provides developers an options to write client side applications using JavaScript in a clean Model View Controller (MVC) way.
    - Applications written in AngularJS are cross-browser compliant. AngularJS automatically handles JavaScript code suitable for each browser.

2.3.2 Other languages for Web application development:

2.3.2.1 Java:

Apache Foundation
FRAMEWORKS
BI
GUI
Java
SOA
UI Frameworks
PERSISTENCE
CONTAINERS

Java is a commonly used language for web development, especially on the server-side. Java web applications are distributed applications that run on the internet. Web development with Java allows us to create dynamic web pages where users can interact with the interface.

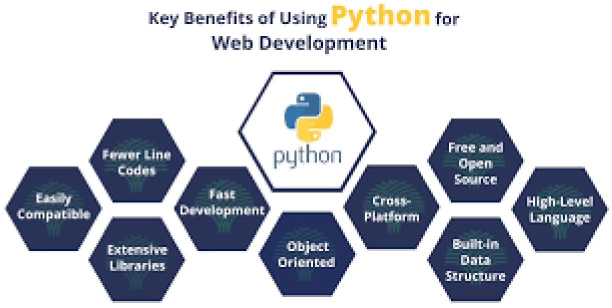
There are various ways through which you can create dynamic web pages in Java. The Java EE (Enterprise Edition) platform provides various Java technologies for web development to developers. Services like distributed computing, web services, etc. are provided by Java EE. Applications can be developed in Java without using any additional scripting language.

A web application helps clients to interact with the server and access information. Java is perfect for developing large web applications too because of its ability to communicate with a large number of systems. Services like peer-web services, database connectivity, and back-end services can also be accessed via Java web development.

Java Web Application Technologies:

1) Servlet
2) JSP
3) JPA
4) JDBC

2.3.2.2 Python(Django) :



Python is one of the most in-demand programming languages. It is because of its ease of use and easy-to-read syntax. As it is a popular and easy language, Python is used in several fields to develop various kinds of applications such as desktop applications, Machine Learning models, and so on. One such field is Web Development.

Python is one of the widely used languages to build web applications. You can use it to perform several tasks; you can even do Web Development by using Python. You can use Python to build web apps in several ways, such as for server-side web apps, RESTful web APIs, etc. As there are so many ways that you can use Python to build web applications, it can be a bit difficult and overwhelming to get started. Hence, in this blog, we will, first, discuss Web Development and then move on to understanding how to use Python for Web Development.

The frontend is the part of a web application that deals with what users can see and interact with and how those interactions should appear. It is what the users see on their web browser when they visit a URL to access a web application. The front-end developer of a web application is concerned with managing interactions with web servers, fetching data, displaying it in the web application, and sending it to the webserver to be saved or manipulated such as a tweet, blog post, etc.

The backend is the part of a web application that deals with its server-side aspects. It is the aspect where Web Development using Python really shines. The backend handles storing, retrieving, and formatting data in an agreed-upon format so that the data can be parsed and understood by other applications. Since, in a web server, data is exchanged via HTTP, the backend of a web application needs to be able to parse an HTTP request, understand its content, and operate accordingly on the data. These operations may include storing, validating, updating, deleting, and retrieving data, and more.

CHAPTER 3: SOFTWARE REQUIREMENT SPECIFICATIONS

"The most important single aspect of software development is to be clear about what

you are trying to build."

3.1 Problem Definition:

A problem statement is a concise description of the problem or issues a project seeks to address. The problem statement identifies the current state, the desired future state and any gaps between the two. A problem statement is an important communication tool that can help ensure everyone working on a project knows what the problem they need to address is and why the project is important.

While solving a coding problem when you try to compile the code then the error was identified during the testing of code or program. There is various condition was given in the problem according to that the user need to solve them. Coding problem is based on the logic that we used to solve that.

Compiler of the IDE will prompt or show the error which occur during testing the code with the location of the error on which line the error are occur.

Importance of Problem:

A problem statement is important to a process improvement project because it helps clearly identify the goals of the project and outline the scope of a project. It also helps guide the activities and decisions of the people who are working on the project. The problem statement can help a business or organization gain support and buy-in for a process improvement project.

1) **Ideal situation:** The first thing your problem statement should describe is what the ideal situation would be if there wasn't a problem you needed to address. This section identifies the goals and scope of the project are. This section should create a clear understanding of what the ideal environment will be once the issue has been resolved.

2) **Reality:** The next section of your problem statement should describe what the current reality is for your company or organization. This section will identify what the problem is, state why it is a problem and identify who the problem is impacting. It will also describe when and where the problem was identified.

3) **Consequences:** The next section of your problem statement should identify what the consequences of the problem are. This section describes the effects of the problem by describing how the people affected by the problem are being impacted and quantifying how much the problem is impacting them. Common consequences can include the loss of time, money, resources, competitive advantage, productivity and more.

4) **Proposal:** The proposal section of a problem statement may contain several possible solutions to the problem, but it is important to remember that it does not need to identify a specific solution. The purpose of the proposal section should be to guide the project team on how they can research, investigate and resolve the problem.

How To Write a Problem statement

1) **Identify the problem:** Before you can begin writing your problem statement, you first need to identify what the problem is.

2) **Begin your statement with your ideal solutions:** Next, you can begin writing your problem statement by describing what the ideal environment would look like if your problem didn't exist. This section should try to describe what your company hopes to accomplish as a result of the process improvement project.

3) **Describe current gaps:** Next, write the reality section of your problem statement. Your goal in this section should be to clearly identify what the current environment looks like. In

this section, you should identify what the problem is, what is causing the problem and why it is an issue. You should also describe when, where and how you were able to identify the problem.

4) **State the consequences of the problem:** Next, write the reality section of your problem statement. Your goal in this section should be to clearly identify what the current environment looks like. In this section, you should identify what the problem is, what is causing the problem and why it is an issue. You should also describe when, where and how you were able to identify the problem.

5) **Propose the addressing the problem:** Finally, end your statement with a proposal section. In this section, you should try to identify how your company will make progress toward reaching your goals and accomplishing your ideal environment. While you may choose to identify several possible solutions in this section, it is more important to focus on identifying how your company will find those solutions than it is to identify the specific solution that will be used.

3.2 Requirement specification:

"Every decent work of software starts by rubbing a developer's itch."

3.2.1 User Requirement:

The User requirements for the innovative system are to make the organization fast, flexible, less inclined to errors and reduce expenditures and save period. A system that can mechanize the examination of answers which are pre-stored that results can be generated as soon as the student gives the reason. A capability that can produce result charts as required. The new system will be more
secure in managing student records and reliable enough to use under all conditions.

The user registers with the log-in authorizations.

The user will be a student, teacher, Employer and administrator.

Administrator can remove the user from the application.

Users can get point of the solving problem.

Points will depend on level of the problem.

Users able to view the no of problem you solve in a day.

Users able to download the data in PDF format.

3.2.2 Functional requirements

Functional requirements define what a product must do, what its features and functions are.

Data structure and functional requirements and as such it has a similar scope and structure. It aims to present an updated compilation of requirements and prioritized recommendations in response to the first evaluations of the authoring tool, the analytics and visualization services, and the 3D virtual labs. First the updated list of metrics which will be incorporated in the shallow and deep analytics services and related visualization tools are presented. Then the update of functional requirements of the various components (authoring tool, analytics and visualization services) in light of the evaluation results from the piloting tests the functional requirements for the combination of Chemistry labs and for an elaborated version of a multi-stage 3D virtual lab/game based on the Wind Energy Lab Finally are presented.

□ The application will work on the internet

□ Data is saved in the database unless and until it is deleted

□ Every record is maintained according to the progress of the student

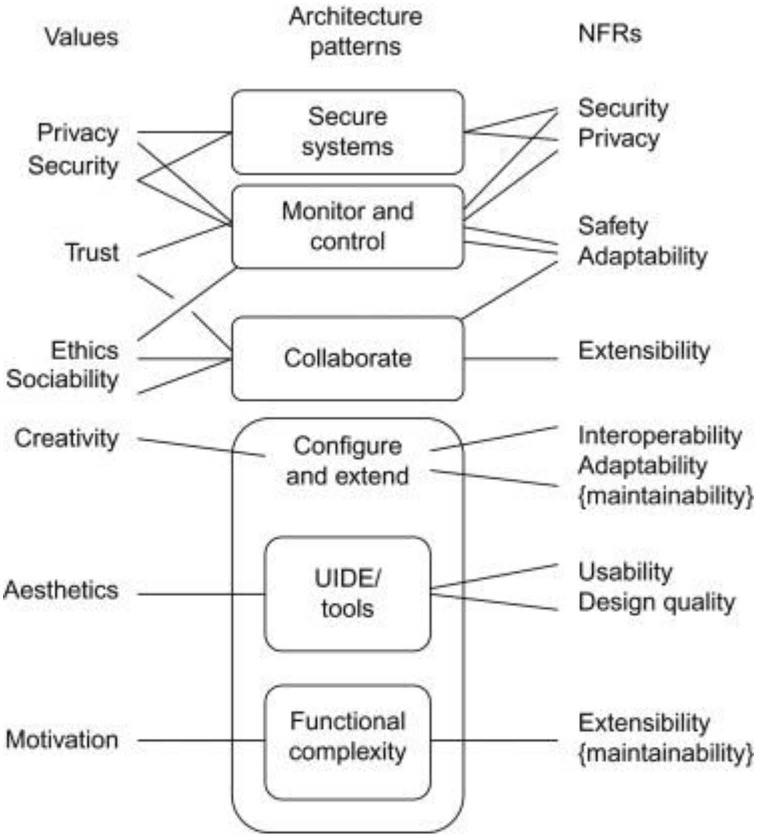
□ Every user, administrator should be registered

□ Every student should have a unique name or id for easy data searching

□ User data must be nursed into the system

3.2.3 Non-functional requirements:

Non-functional requirements (also known as *quality requirements*) are generally more difficult to express in a measurable way, making them more difficult to analyze. In particular, NFRs tend to be properties of a system as a whole, and hence cannot be verified for individual components. Recent work by both researchers and practitioners has investigated how to model NFRs and to express them in a form that is measurable or testable. There also is a growing body of research concerned with particular kinds of NFRs, such as safety, security, reliability, and usability.



.

<u>Maintainability</u>: The application will maintain data like:

Username

Email address

Password for authentication

Administrator username and password

Percentage of the solving problem

3.4 Software & Hardware Requirements

3.4.1 Hardware requirements:

Frameworks bought inside the previous years will ordinarily be equipped with the suggested highlights. It would be ideal if users support that smartphone meet these prerequisites. Recall that if a user's mobile phone doesn't meet the necessities, among different issues, keeping it accurately designed when the user experiences issues.

To create the product:

Processor: Minimum Intel Core i3 6th generation or Intel Core i5-5200u CPU@2.20GHZ

2.20GHZ.

Memory: 4GB RAM for faster speed we can use 8GB RAM.

System Type: 64bit Operating System.

Display: 1280 x 800 Least Screen resolution

Graphics: Intel Integrated HD Graphics 520.

HARD DISK: Minimum 60 GB available free disk space.

To use the application:

Operating system: Windows, MacOS, Android 4.1, 4.2, and more

Storage: Between 2 GB and 8 GB

Memory: At least 2 or 3 GB RAM to run without any risks

Display: 1280 x 800 pixels or higher on a 10-inch device

Internet: This application is designed to work online. A high-speed Internet connection is recommended.

3.4.2 Software requirements

Front end: HTML, CSS, JavaScript, React.js, React Reveal

Back end: Node.js, MongoDB

Google chrome or any other browser stable release

Testing tools according to the project

3.5 Planning and Scheduling

Planning is about electing and designing operative policies and methodologies to accomplish project objectives. Scheduling is procedure of transmission tasks to get them accomplished by allocating suitable resources within an estimated budget and time frame.

Planning for a framework is finished by the suitably named scheduler, which is for the most part worried about three things:

Throughput or how quick it can complete a specific number of assignments from start to finish per unit of time

Latency is the turnaround time or the time it takes to complete the undertaking from the hour of solicitation or accommodation until the finish, which incorporates the holding up time before it could be served

Response time which is the time it takes for the procedure or solicitation to be served, in short, the holding uptime

Planning is generally founded on the elements referenced above and differs relying upon the framework and the programming of the frameworks or client's inclinations and goals. In present-day Smartphones, for example, Smartphones with a portion of handling power and different assets and with the capacity to perform various tasks by running numerous strings or pipelines on the double, planning is not, at this point a major issue and most occasions procedures and applications are sans given rule with additional assets, yet the scheduler is still working diligently overseeing demands.

1) **Plan schedule management**: This document also establishes who has the authority to make schedule changes, the process team members should follow in order to request a change, and a project communication plan to alert the team of changes made during the course of the project.
2) **Define the project activities:** In team task management, tasks requiring fewer than eight hours could be grouped with others and tasks over eighty hours are likely too cumbersome and should be broken down further. Activities should also be measurable, easily estimated, and related to both a project deliverable and a budgeted cost.
3) **Determine dependencies:** If you're building a house, for example, you can't put the roof on until the frame is completed. It's important to correctly define all your project dependencies so you can schedule accurately and avoid project delays.
4) **Sequence activities:** After you've established dependencies among your activities, you can sequence them. At this point, you aren't assigning any time to your activities in terms of work hours or due dates. Instead, you're focusing on the order in which all project activities should be done so that the most efficient flow is created.
5) **Estimate resources:** Each activity in your project will require resources in the form of personnel, subcontractor costs, tools (physical and/or digital tools like software programs), and workspace. Make sure to consider other resources that are specific to your industry or project. Estimate the resources needed for each project activity.
6) **Estimate durations:** Overestimating could leave team members or other resources sitting idle as they wait for antecedent tasks to be completed. The best way to estimate duration is to use data from similar previous jobs. If you don't have any data to work from and there's no industry standard to which you can refer, an estimate based on the average of the best, worst, and most likely scenarios.

7) **Develop the project schedule:** There are multiple models and formulas for developing the project schedule, including critical_path, critical chain, and resource leveling among others. Each of those methods is worthy of an article in itself, so we won't cover them here. Take the time to find a method that works well for you. For example, don't ignore the calendar! Check vacation requests from team members. Don't forget to include factors like national holidays, corporate functions, stakeholder events, and other occasions that may affect your schedule. If the whole company shuts down for a holiday week, you'll need to add that time to your due dates and manage customer expectations accordingly.

8) **Monitor and Control:** When schedule changes must be made, you ensure they are carried out and communicated according to the plan laid out in Step 1. Throughout the project, you will ensure that each activity is on schedule and determine whether corrective action needs to be taken if delays occur.

CHAPTER 4: SYSTEM DESIGN

“A design system acts as the connective tissue that holds together your entire platform.”

4.1 INTRODUCTION:

Systems design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. It is the process of defining, developing and designing systems which satisfies the specific needs and requirements of a business or organization.

A systemic approach is required for a coherent and well-running system. Bottom-Up or Top-Down approach is required to take into account all related variables of the system. A designer uses the modelling languages to express the information and knowledge in a structure of system that is defined by a consistent set of rules and definitions. The designs can be defined in graphical or textual modelling languages.

Some of the examples of graphical modelling languages are:

- Unified Modelling Language (UML): To describe software both structurally and behaviorally with graphical notation.
- Flowchart: A schematic or stepwise representation of an algorithm.
- Business Process Modelling Notation (BPMN): Used for Process Modelling language.

- Systems Modelling Language (SysML): Used for systems engineering.

Design methods:

- Architectural design: To describes the views, models, behavior, and structure of the system.
- Logical design: To represent the data flow, inputs and outputs of the system. Example: ER Diagrams (Entity Relationship Diagrams).
- Physical design: Defined as a) How users add information to the system and how the system represents information back to the user. b) How the data is modelled and stored within the system. c) How data moves through the system, how data is validated, secured and/or transformed as it flows through and out of the system.

4.2 Basic Modules

DSA Cracker application will consist of the following parts:

Login Module:

There will be two types of users in this system like Student (Teacher or Employer) and administrator to perform any task in this application. The student will have to login through username or email and password for authentication but user will not have permission to remove or add anyone from this application. The administrator has all the permission to add or remove the user from this application.

This will contain the user photo or logo and information of the user as per the user provided like:

- Name
- Date of birth
- Gender
- Username
- Email

- Phone Number
- Qualification

Home Page Module:

There is different section in the homepage like array, matric, string, search & sort, linked list, binary trees, BST, greedy, backtracking, stacks & queues, heap, graph, trie, dynamic programming, bit manipulation in which there are certain number of questions to solve. To solve the problem, you need to click on the section in which there are button of “start now” click on it to enter the list of the question.

Reset Progress:

This option will reset your progress to start point like from 0 (zero).

Export Progress:

This option will download the file in json format to show the question you visit or solve ethe problem.

Import Progress:

This option will import all the file or solution of the question into the section according to the format of the question.

4.3 Procedural Design

4.3.1 Flowchart diagram

Flowchart:

A flowchart is a diagram that depicts a process, system or computer algorithm. They are widely used in multiple fields to document, study, plan, improve and communicate often complex processes in clear, easy-to-understand diagrams. Flowcharts, sometimes spelled as flow charts, use rectangles, ovals, diamonds and potentially numerous other shapes to define the type of step, along with connecting arrows to define flow and sequence. They can range from simple, hand-drawn charts to comprehensive computer-drawn diagrams depicting multiple steps and routes. If we consider all the various forms of flowcharts, they are one of the most common diagrams on the planet, used by both technical and non-technical people in numerous fields. Flowcharts are sometimes called by more specialized names such as Process Flowchart, Process Map, Functional Flowchart, Business Process Mapping, Business Process Modeling and Notation (BPMN), or Process Flow Diagram (PFD). They are related to other popular diagrams, such as Data Flow Diagrams (DFDs) and Unified Modeling Language (UML) Activity Diagrams.

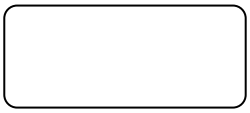
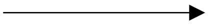
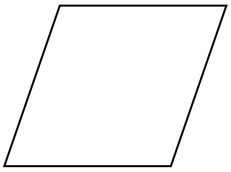
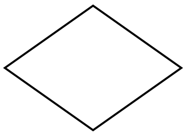
Data objects showed by tagged arrows or transformation right-angled degree outlined by circles known as bubble and DFD is given as grading fashion that's the main knowledge flow model outlined the system as an entire resultant DFR routine context drawing provides cumulative details with every successive level.

Rules of DFD diagram:

- Maintain the scope of the system suggests that of context diagram.
- Maintain DFD so main consecutive sequence of the actions.
- Acknowledge all inputs or outputs.
- Each process must have minimum one input and output.
- Each data store must have minimum one data flow in and one data flow out.
- Data warehoused in a system must go through a process.

A DFD shows the flow of data through a system. It views a system as a -function that transformed. The inputs into desired outputs. Any complex system will not perform this transformation in a only Step and data will typically under decodes of transformation before it becomes the output - for types of date entity are distinguished in a data flow diagram:

Name	**Symbol**	**Description**

Start/end		An oval represents a start or end point
Arrows		**A line is a connector that shows relationship between the relationship shapes.**
Input/Output		**A parallelogram represents input or output.**
Decision		**A diamond indicates a decision.**

The most common symbol used in a flowchart is the rectangle. A rectangle represents a process, operation, or a task. The next most common symbol is the diamond which is used to represent a decision. The decision shape is represented as a Diamond. This object is always used in a process flow to ask a question. And, the answer to the question determines the arrows coming out of the Diamond. This shape is quite unique with two arrows coming out of it. One from the bottom point corresponding to Yes or True and one from either the right/left point corresponding to No or False. The arrows should always be labeled to avoid confusion in the process flow. This object is represented by a rectangle with the top sloping up from left to right. The Manual Input object signifies an action where the user is prompted for information that must be manually input into a system. This shape takes two names – ‘Subroutine’ or ‘Predefined Process’. Its called a subroutine if you use this object in flowcharting a software program. This allows you to write one subroutine and call it as often as you like from anywhere in the code. The same object is also called a Predefined Process. This means the flowchart for the predefined process has to be already drawn, and you should reference the flowchart for more information. Although these are the standard symbols available in most flowchart software, some people do use different shapes for different meanings. The most common example of this is using circles to denote start and end. The examples in this flowchart tutorial will stick with the standard symbols. Although you can start drawing flowcharts by scratch it is much easier to use templates. They help you reduce errors and remind you about the best practices to follow. If you want to use a ready-made template, go to the flowchart examples section and click on the flowchart that best suits you. Click on the use at template after the image and you are ready to draw your flowchart. Also commonly known as the flowchart input/output symbol, it represents information or material that's received — a flowchart input symbol — or generated like an e-mail or physical product — a flowchart output symbol. Same symbol whether it's an input or output. This simple circle represents the continuation of flow through multiple charts or pages. Used in more complex charts, the connector usually contains the same letter or number where the flow breaks in one chart or page and where it continues in another. Sometimes you need a little more explanation, the note symbol provides space for an explanation and/or comments within a specified area of a diagram. Typically, it's connected by a dashed line.

S

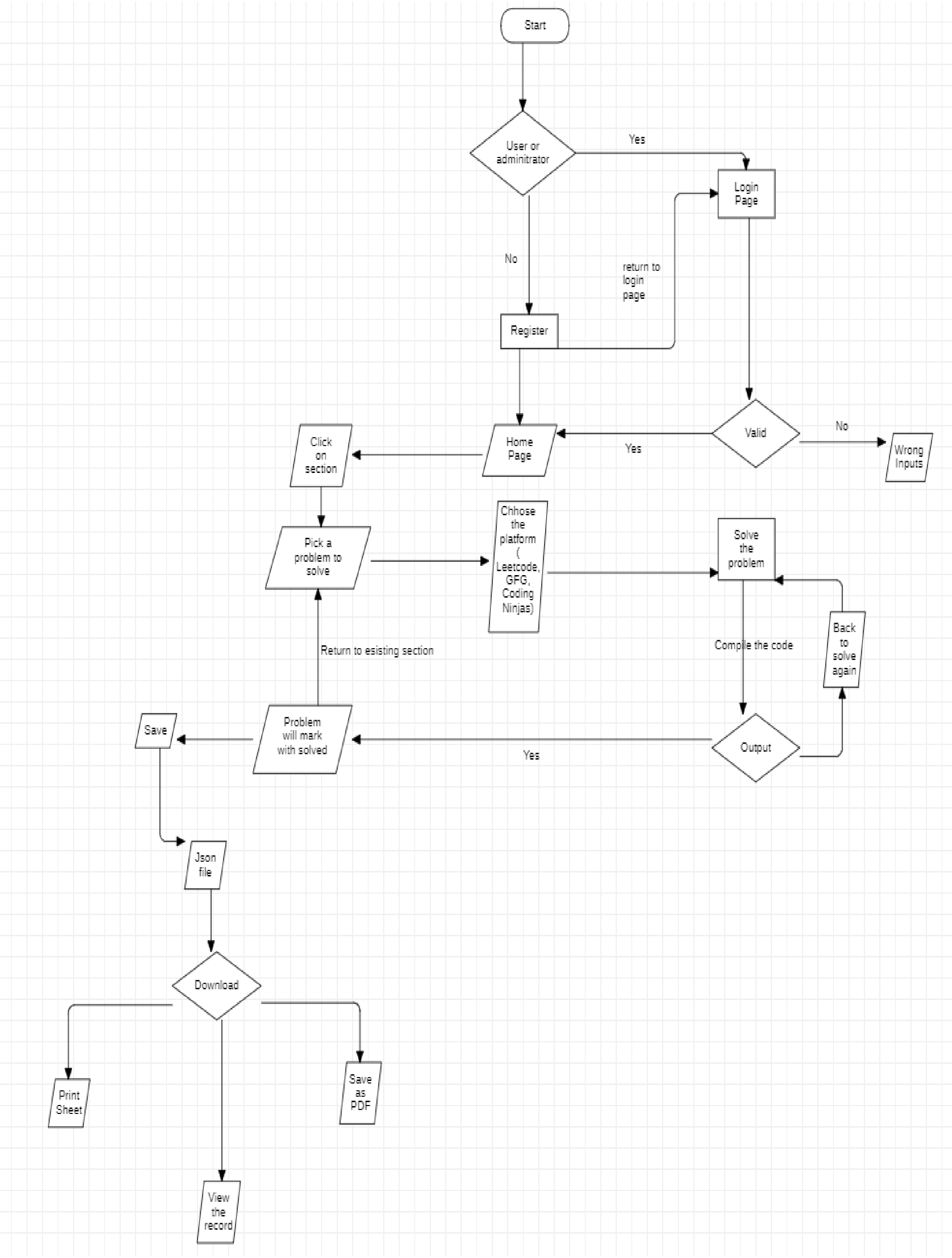
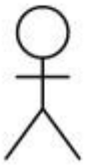
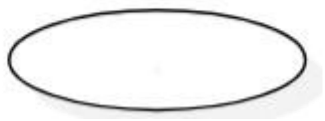


Fig. 4. 1Flowchart Diagram

4.3.2 Use case diagram

In the Unified Modelling Language (UML), a use case diagram can recapitulate the particulars of your system's users (also known as actors) and their connections with the system

The main perseverance of the usecase diagram is to describe the operational aspect of the system. It accumulates system requirements, including both internal and external influences. It calls people, use cases, and many things that call out the actors and elements responsible for implementing use case diagrams. It characterizes how an entity from the external environment can cooperate with a part of the system.

Name	**Symbol**	**Description**
Actor		An actor signifies a user or another system that will interrelate with the system you are demonstrating
Use case		A use case is an exterior view of the system that signifies some action the user might perform to whole a task.

Use case diagram purpose:

- It collects the system requirements.
- It shows external view of system.
- It finds the interior and exterior factors which influence the system.
- It shows interaction between actors.

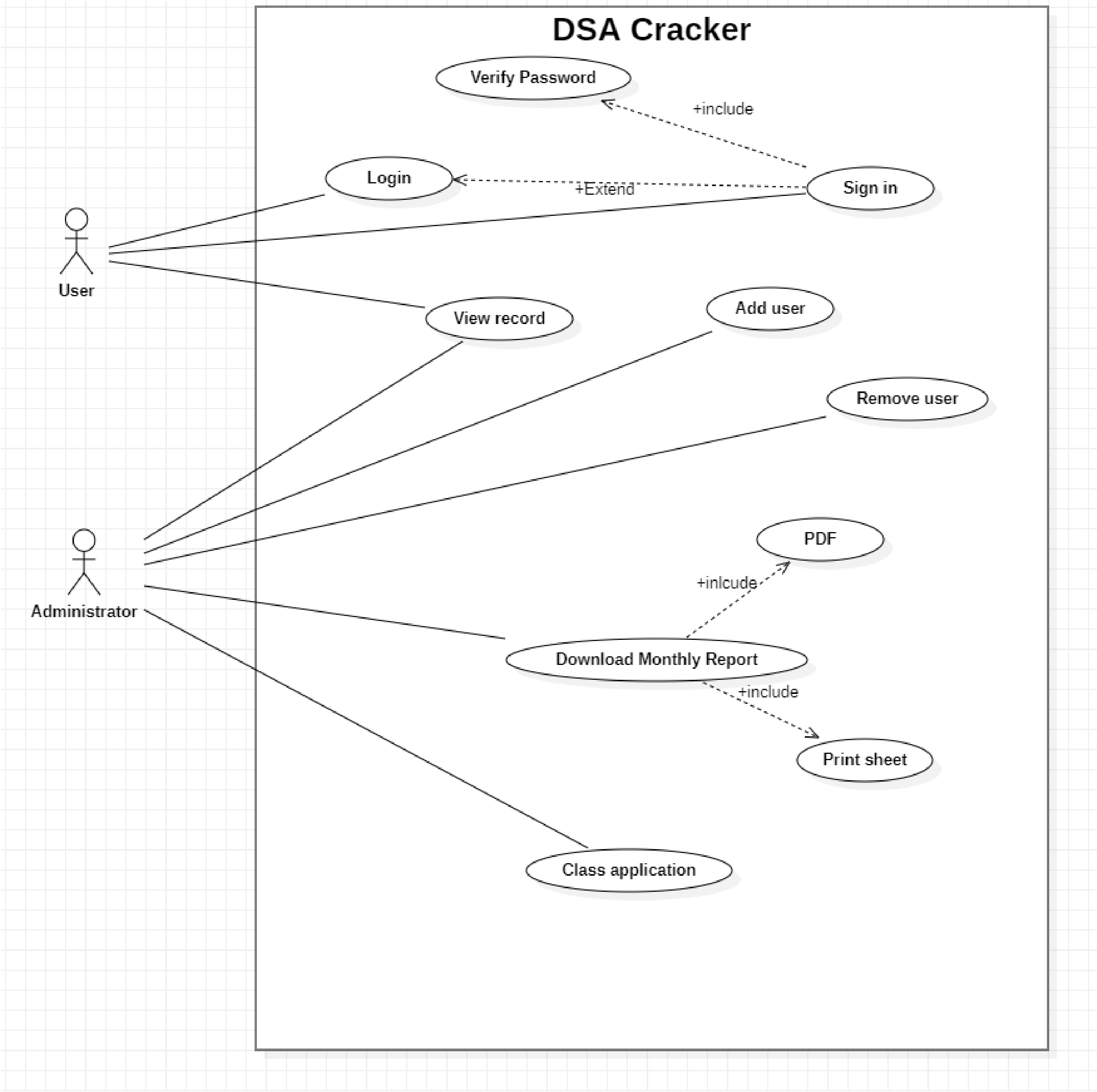


Fig. 4. 2 **Use case diagram**