CHAPTER 1
INTRODUCTION

1.1 Problem Definition

The initial investigation is conducted in order to gather the problems that are currently faced by existing systems and started addressing the common problems and challenges that common users facing related to their career. Initially, the only way to choose the appropriate career is based on the information that they know or based on what others say. This project made an attempt to develop a system which is economical, time-saving and helps the users to identify the best career suited for them.

The issue for developing this web application was how to meet the requirements of different career options and different career options. The issues are divided into sub-problems to well- define the limitations and scope of the project topic.

1.2 ABOUT THE ORGANIZATION

Infosys was established by seven engineers in Pune, Maharashtra, India with an initial capital of $250 in 1981. It was registered as Infosys Consultants Private Limited on 2 July 1981. In 1983, it relocated its office to Bangalore, Karnataka, India.

Infosys Ltd is a global technology services firm that defines, designs and delivers information technology (IT)-enabled business solutions to their clients. The company provides end-to-end business solutions that leverage technology for their clients, including technical consulting, design, development, product engineering, maintenance, systems integration, package-enabled consulting, and implementation and infrastructure management services.

The company also provides software products to the banking industry. They have developed Finacle, a universal banking solution to large and medium size banks across India and overseas. Infosys BPO is a majority owned subsidiary. Through Infosys BPO, the company provides business process management services, such as offsite customer relationship management, finance and accounting, and administration and sales order processing. The company is having marketing and technical alliance with FileNet, IBM, Intel, Microsoft, Oracle and System Application Products.

Infosys Ltd is a public limited and India's second largest software exporter company was incorporated in the year 1981 as Infosys Consultants Pvt Ltd by Mr. N.R. Narayana Murthy at Karnataka. The company was started by seven people with the investment of USD 250. The company became a public limited company in the year 1992. The company was the first Indian company to be listed on the NASDAQ in the year 1999. Infosys also forms a part of the NASDAQ-100 index. Continuously in the year 2001, 2002 and 2003, the company wins the National award for excellence in corporate governance conferred by the Government of India.

1.3 OBJECTIVE OF THE PROJECT

CareerAccelerate is an online website which is mainly focuses on users who are confused regarding which career is best for them. In this application the registered users are categorically divided into three 10^{th},12^{th} and non-age specific. So for each user belonged to their specific category they have to take an assessment. For each category there are different questions that will cover the different areas like, for 10^{th} assessment the questions generally cover four areas they are:- Likes/Dislikes, Career based situations, Aptitude, General Awareness and for 12^{th} assessment it will cover the areas like:- Situational Exploration, Logical Scenarios, Visual Logic and for non-age the assessment questions will cover the areas like Self-Identification, Career based situations, Importance Factors, Situational Exploration, Logical scenarios. These questions then evaluate the thinking skills, reasoning ability and the aptitude level with the help of psychometric algorithm which will predict which career is best for them. The application, based on the answers provided by users in any of the above assessments, will generate a statistical report, that would provide a detailed view of the best suited career options for them. The application should incorporate a psychometric analytical algorithm for this purpose. It should allow the users to download the generated report in a PDF format.

.

CHAPTER 2

LITERATURE SURVEY

2.1 ININTIAL INVESTIGATION

The initial investigation is conducted in order to gather more information about the existing system. The important thing that a user who are the part of application needs is the best career for them. For this application the important thing is to identify the different users in different category and how to predict the best career for them with good accuracy. For that we categorized the users into three and prepared different questions for each category to identify their various skills like aptitude, technical skills etc… and also for getting good prediction with high accuracy we used psychometric analysis algorithm.

2.2 EXISTING SYSTEM

There are some applications that are implemented for career guidance. But they are not using any specific algorithms in their application also they will not get adapted to any other screen sizes

2.3 PROPOSED SYSTEM

Many career guidance Systems have been developed based upon different platforms and concepts. Use of career related applications are growing but there are many issues related to their accuracy in prediction. ”CareerAccelerate” is a career guidance that help the users who are confused regarding which career is most suitable for them. UI will adapt to the different screen sizes and resolutions. Where the existing system will not adapt different resolutions. When comparing accuracy of the existing system with our application, the accuracy of CarrerAccelerate is high because of the usage of algorithms.

In this application we are using Psychometric algorithm. **Psychometrics** is a field of study concerned with the theory and technique of psychological measurement. The field is concerned with the objective measurement of skills and knowledge, abilities, attitudes, personality traits, and educational achievement. Some psychometric researchers focus on the construction and validation of assessment instruments such

as questionnaires, tests, raters' judgments, and personality tests. Others focus on research relating to measurement theory. In this application we are using one of the model called Big Five personality Traits model(Ocean model).

The personality model which has garnered immense popularity in last 20 years or so is the Big Five Personality Traits Model also known as Five Factor Model of Personality or the OCEAN Model. According to the OCEAN Model, there are only five main components of personality. These five components are **O**penness, **C**onsciousness, **E**xtraversion, **A**greeableness and **N**euroticism. As you can understand, OCEAN is the acronym of the five main traits in the model.

Fig 2

Following is a detailed explanation of Big Five Personality Traits Model and how it can help you better understand people and their behaviors.

Openness

This is the first and the most important trait in the OCEAN model of personality assessment. Insight and imagination are some of the major characteristics of this trait. If you have this trait, you usually have a dynamic personality and wide range of interests. You are always willing to explore the world. You are curious about other things and want to find out more about other people. Similarly, you are very keen to enjoy new experiences and learn new things.

People also tend to be more creative and bold if they have this trait. People high on open continuum are usually imaginative and creative whereas those low on this continuum are usually conventional and down-to-earth. Similarly, “open” people have the tendency to hold unusual beliefs and they are sensitive to beauty and logically curious. They love to take on new challenges as well.

On the other hand, you may find it difficult to think abstractly and are more on the traditional side of things if you are low on this trait. You dislike change and do not enjoy new things or want to learn them. You are not very imaginative and as a result, resist new ideas as well. You deride theoretical or abstract concepts. You normally discard novelty in favor of familiarity. You may have no desire to change and are happy with your current lifestyle if you are not an open person.
Typical example of an open person would be of someone who spontaneously decides to visit new places. It means he is willing to embrace new ideas and open to new experiences. Reserve example would be of a person who occupies himself with his daily routines and does not want to experience new things.

Consciousness

If you are high on this extremely important personality trait, you are actually a goal-oriented person. You have exceptional pulse control with highest levels of thoughtfulness. Similarly, mindfulness to details and exceptional organizational abilities are main characteristics of highly conscientious people. They are extremely mindful of deadlines and constantly think about how their behaviors and actions are affecting people. Similarly, they plan everything well in advance and organize their lives accordingly.
A highly contentious person is also highly disciplined, deliberate and careful. Contentiousness is also a measure of a person's productivity, especially of low level employees in organizations. Such persons are better planners as compared to open ones who prefer to live their lives more freely. They are dutiful and self-disciplined. In fact, they spend a lot of time planning and preparing for their next tasks. They like to follow a set schedule and try to finish important tasks right away.

People low on this particular trait often exhibit totally contrasting behaviors. They are more impulsive and therefore, deride schedules, planes and structures. They usually do not care about most things and have the tendency to make messes. They find it very difficult or even impossible to rectify their mistakes. They fail to put things back to where they belong. Unlike highly conscientious persons, they are adept in procrastinating and seldom complete important tasks in time. Moreover, they seldom do anything in life which they are expected or supposed to do.

A person who always plans his morning meeting in advance and stick to the agenda in the actual

meeting can be an example of a highly conscientious person. On the other hand, if a person does no planning and has no idea of what to do in the day both at his work and home is an example of a typical procrastinator. Such people are low on conscientiousness and always make mess of even the most basic things in their lives.

Extraversion

If a person is emotionally expressive and likes to assert his ideas on others, he happens to be an extravert or extrovert. Such people also tend to be very talkative and characteristically social and excitable. Being in social situations gives them more energy and courage to go about their business. As expected, extraverts are very outgoing and they derive energy and excitement by talking to other people and sharing their ideas with them.

Extroverts are highly energetic, enthusiastic and easily visible to people around them. They like to take action and see results no matter whatever it takes. They are often the conversation starters in social gatherings and like to be the center of attraction as well. As a result, they also have a wide circle of friends and acquaintances. Similarly, they befriend new people rather easily. However, they normally do not think before speaking which sometimes undermine their personality. Extraverts tend to be better leaders as well.

Opposite to extraversion is introversion. Introverts like to keep to themselves. They do not like to be a part of the social world. They love solitude, tend to be low-key and get bored and exhausted whenever they have to socialize. Similarly, they are also not good conversation starters and tend not to become part of any discussion as well. However, they carefully think before saying anything and dislike small talks. Most importantly, they do not want to become the focal point of any social gathering.

It was commonly believed that introverts do not become good leaders but recent studies suggest they can do equally good as extraverts when it comes to leadership.

Typical example of an extravert would be of a person who loves to party and going out with friends. Reverse example would be someone who feels less energetic and exhausted even on the thought of any social engagement. He rather likes to submerge himself in a pile of books in a locked room than attending a party or office meeting

A**greeableness**

Agreeable people are some of the best people you would ever meet. They are affectionate, kind, full of empathy for others and extremely trustable. They are very helpful, generous and considerate. They can even compromise their own interests in order for you to get what you want. Altruism is another extremely important characteristic of agreeable people. These are the most cooperative people both in your personal and professional lives. They get along and interact with others nicely, making them an indispensible part of any team as well.

People high on this important personality trait often like to volunteer and engage in pro-social behaviors. Similarly, such people take keen interest in other people lives. They contribute heavily to make them happy and live a better life. They never hesitate to help those who are in trouble. Agreeable people try their level best to avoid negative thoughts and behaviors and live a happier life as a result. One of the most interesting studies related to personality traits also suggests that people having a looser gate tend be less conscientious and more agreeable.

People who are low in agreeableness prefer their own self-interest over anything else. They have no concern for others and always try to mind their own business. They are normally unfriendly and will never compromise their interests for the welfare of others. They take little to no interest in solving other people's problems and have no regard for their feelings. They often look down upon people in distress. Just like people with narcissistic personality disorder, they manipulate others to grind their own axe. They also never hesitate to bully and insult others.

For example, a person is agreeable if he takes time out of his extremely busy schedule to attend his friend's wedding. On the other hand, Sherlock Homes is the classic example of people who don't bother about being agreeable. They take no interest in others' problems. They even feel insulted if someone calls them for help because they never want to compromise their own interests.

Neuroticism

Neuroticism is the last and yet another important trait of the Big Five Personality Traits Model. Some of the important characteristics of neuroticism are emotional instability, sulkiness and

unhappiness. People having high levels of neuroticism are easily irritable and experience frequent bouts of anxiety, sadness and mood swings. Therefore, neuroticism is also a person's ability to counter stress in a complex or difficult situations.

Neuroticism may also lead to many psychological problems including a lot of stress. You start worrying about a lot of things, most of which do not really matter. You always feel anxious and get upset rather easily even by trifles. People with extreme case of neuroticism also find it very difficult to recover from a period of high stress and anxiety.

If you have a high score of neuroticism, you are very likely to suffer from depression, anger, frustration and exhibit many other negative behaviors. A high score also means you are temperamental, very self-centered and you vary between emotions and feelings a lot. You might often have constant feeling of insecurity as well. Some people correlate Freudianism and neuroticism. They do have many similarities but they are not identical in any respect.

As you might have guessed, emotional stability is the opposite of neuroticism. People with high levels of emotional stability are emotionally calm and deal with stress rather successfully. They always remain relaxed regardless of the situation and seldom feel depressed or sad. They do not worry much and always see positive side of things.

You can find examples of people with high scores of neuroticism in everyday life. For instance, if a person shouts at a waitress for too much sugar in his coffee or not getting his order in time, he is likely to have high levels of neuroticism. On the other hand, a person is low on neuroticism if he calmly handles even the worst of criticisms in the office or anywhere else.

2.3.1User Classes and Characteristics

The system contains three users, admin, registered user, guest user. The admin is the system itself. The system will provide the assessment based on their category and will generate the report based on the assessment. The guest users are not the part of this application they can just view the basic information's in this application. They can become the part of the application only after the successful registration. After successful registration they become registered user. Registered users are part of the application,

they have a username and password for login. After successful login they can take the Assessment based on their category. There are basically three categories based on their current education they are 10th,12th, & non-age. The result of the assessment will be displayed in a tabular form and they can download it.

2.4 FEASIBILITY STUDY

During system analysis, a feasibility study of the proposed system was carried out to see whether it was beneficial to the users. The main aim of the feasibility study is to determine whether it would be financially and technically feasible to develop the product. While evaluating the existing system, many advantages and disadvantages raised. Analyzing the problem thoroughly forms the vital part of the system buddy. Problematic areas are identified and information is collected.

The benefits of this site are users can easily interact and get the services without much complexity. It helps to make it possible that more users can interact with the site at a time. Feasibility study is to determine whether the proposed system is technically, economically and behaviorally feasible in all respects.

The main aim of feasibility study is to evaluate alternative site and propose the most feasible and desirable site for development. If there is no loss for the organization, then the proposed system is considered financially feasible. A feasibility study is carried out to select the best system that meets performance requirements. The feasibility study activity involves the analysis of the problem and collection of all relevant information relating to the product such as the different data items which would be input to the system, the processing required to be carried out on these data, the output data required to be produced by the system as well as various constraints on the behavior of the system.

In this scenario, problems are identified. Essential data are being gathered for the existing problems. It is necessary that this analysis familiarizes the designer with objectives, activities, and the function of the organization in which the system is to be implemented. The feasibility study was divided into four: - Technical, Economical, Operational and Behavioral. It is summarized below: -

2.4.1 Technical Feasibility

According to feasibility analysis procedure the technical feasibility of the system is analyzed and the technical requirements such as software facilities, procedure, inputs, are identified. While considering the problems of existing system, it is sufficient to implement the new system. The proposed system can be implemented to solve issues in the existing system. It includes the evaluation of and how it meets the proposed system. This system use visual studio 2010 as front end technology and oracle as back end technology.

2.4.2 Economic Feasibility

Economic analysis is most frequent used for evaluating of the effectiveness of the candidate system. More commonly known as cost/benefit analysis the procedure is to determine the benefit and saving that are expected from a candidate system and compare them with the existing system. Except for the initial capital amount and the amount after each financial year, no other huge amount is needed. The expenses can be handles by any participants. So, the system is economically feasible.

This feasibility involves some questions such as whether the firm can afford to build the system, whether its benefits should substantially exceed its costs, and whether the project has higher priority and profits than other projects that might use the same resources. Here there is no problem. This firm has fully equipped hard ware, and fully fledged software, so no need to spend money on these issues. And as the client and the developer are one, there is no further problem in economic issues.

2.4.3 Operational Feasibility

Methods of processing and presentation are all according to the needs of clients since they can meet all user requirements here. The proposed system will not cause any problem under any circumstances and will work according to the specifications mentioned. Hence the proposed system is operationally feasible. People are inherently resistant to change and computer has been known to facilitate changes. The system operation is the longest phase in the development life cycle of a system. So, Operational Feasibility should be given much importance. This system has a user-friendly interface. Thus it is easy to handle.

2.4.1 Behavioral Feasibility

In today's world, computer is an inevitable entity. As per the definition of behavior design, many valid points are recognized in this study. This system behavior changes according to different environment. In order to ensure proper authentication and authorization and security of sensitive data of the admin or employers, login facilities are provided. These are the main feasibility studies tested in this application

Chapter 3

SYSTEM ANALYSIS AND DESIGN

3.1 SOFTWARE REQUIREMENT SPECIFICATION

3.1.1 Project Scope

System Requirements Specification is a structured collection of information that incorporates the requirements of a system. This gives an idea about the system specifications required to develop and install the project "CarreerAccelerate". The System Requirements Specification is a technical specification of requirements for the software product. The goal of software requirements definition is to completely and consistently specify the technical requirements for the software product in a concise and unambiguous manner.

The System Requirements Specification is based on the System Definition. The requirement specifications are primarily concerned with functional and performance aspect of a software product and emphasis are placed on specifying product characteristics implying how the product will provide those characteristics. One of the most difficult tasks is selecting software, once the system requirement is find out then we have to determine whether a particular software package fits for those system requirements. This selection summarizes the application requirement.

3.1.1 Product Perspective

The application incorporate a user-friendly design. It will cater to the needs of users looking for help in terms of their career. They allow user to create a profile of their own. Also, the users should be categorically divided based on their present education and the career path for which they are planning to seek guidance for which, users to need key-in specific details about their personal and professional background. The application incorporate a psychometric data analysis algorithm that will help users find the best suited career path for themselves.

3.2.1. FUNCTIONAL REQUIREMENTS

The Application contains the following features: It will give a brief information about CareerAccelerate and its business utility and it will provide an option in the homepage for users to register and login. In this application the users are categorically divided as following:

- 10th Standard or Equivalent passed-outs
- 12th Standard or Equivalent passed-outs
- Non-Age Specific

It will provide the following types of psychometric assessment plans to choose from:

- DriveRight-DriveNow: A Path Selector (Only for 10th Standard or Equivalent passed-outs in order to help them choose the best career ahead). 'A Path Selector' may have the following categories of questions: Likes/Dislikes,Career based situations, Aptitude, General Awareness
- Science-o-Pedia: An Engineer's Choice (Only for 12th Standard or Equivalent passed-outs in order to help them choose the most appropriate stream of Engineering for themselves). Science-o-Pedia: An Engineer's Choice' may have the following categories of questions: Situational Exploration , Logical Scenarios ,Visual Logic
- CareerDoneRight: The Best Way for You (Non-Age Specific in order to help them choose the best career path/profession).'CareerDoneRight: The Best Way for You' may have the following categories of questions: Self-Identification, Career based situations, Importance Factors, Situational Exploration, Logical Scenarios

The application, based on the answers provided by users in any of the above assessments, will generate a statistical report, that will provide a detailed view of the best suited career options for them. The application will incorporate a psychometric

analytical algorithm for this purpose and It will allow the users to download the generated report in a PDF format. Application have the following user types:

- Guest users
- Registered Users

When a guest user opens the application he/she should have the following options: Viewing basic information about CareerAccelerate and its business utility. They can Register as a User by providing the required details. When a registered User opens the application, he/she have the following options:

- Should be able to maintain a profile
- Should be able to take up any of the available assessments based on the category he/she belongs to
- Should be able to view and download the statistical report generated from the application in PDF format

3.2.2. NON- FUNCTIONAL REQUIREMENTS

3.2.2.1Performance Requirements

Website should be light and responsive

3.2.2.2Security Requirement

Users should be able to login by providing a unique user id and password which will fulfil the security requirements.

3.2.2.3Multi language Support

Should support only English language.

3.2.2.4 Scalability

The website must support multiple users without performance degradation.

3.2.2.5Availability / reliability

The probability of failure should be as less as possible

HARDWARE AND SOFTWARE REQUIREMENTS

Deployment Environment Requirements

Hardware requirements

Processor/RAM/HDD : Intel Core Processor / 3GB RAM / 100GB HDD

Web server : Internet Information Server

Database Server : SQL Server 2005

Software requirements

OS for Web server : Windows 7

OS for Database Server : Windows 7

DBMS : Oracle SQL+

Development Environment Requirements

Hardware requirements

Processor/RAM/HDD : Intel Core Processor / 2GB RAM / 100GB HDD

Web server : Internet Information Server

Database Server : SQL Server 2005

Software requirements

OS for Web server : Windows 7

OS for Database Server : Windows 7

DBMS : Oracle SQL+

IDE : Visual Studio 2010

3.3 UML DIAGRAM

UML is a way of visualizing a software program using a collection of diagrams. The notation has evolved from the work of Grady Booch, James Rumbaugh, Ivar JAcobson

and the Rational Software Corporation to be used for object-oriented design, but it has since been extended to cover a wider variety of software engineering projects. Today, UML is accepted by the Object Management Group(OMG) as the standard for modelling software development.

UML stands for Unified Modeling Language. UML 2.0 helps extend the original UML specification to cover a wider portion of software development efforts including agile practices. Improved integration between structural models like class diagrams and behavior models like activity diagrams. The original UML specified nine diagrams; UML

2.x brings that number up to 13. The four new diagrams are called: communication diagram, composite diagram, interaction overview diagram and timing diagram. It also renamed state chart diagrams to state machine diagrams, also known as state diagrams.

Types of UML diagrams

The current UML standards call for 13 different types of diagrams: class, activity, object, use case, sequence, package, state, component, communication, composite structure, interaction overview, timing and deployment. These diagrams are organized into two distinct groups: structural diagrams and behavioral or interaction diagram.

Structural UML diagrams

- Class diagram
- Package diagram
- Object diagram
- Component diagram
- Composite structure diagram
- Deployment diagram

Behavioral UML diagrams

- Activity Diagram
- Sequence diagram
- Use case diagram
- State diagram
- Communication diagram
- Interaction overview diagram
- Timing diagram

3.2.1 Usecase Diagram

To model a system the most important aspect is capture the dynamic behaviour. To modify a bit in details, dynamic behaviour of the system when it is running or operating. So only behaviour is not sufficient to model a system rather dynamic behaviour is more important than static behaviour. In UML there are five diagrams available to model dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature there should be some internal or external factors for making the interaction. These internal and external agents are known as actors. So use case diagram consists of actors, use case and their relationships. The diagram is used to model the system of an application. A single use case diagram captures a particular functionality of a system.

Use case Diagram objects:

- Actor
- Use case
- System
- Package

Actor

Actor is a use case diagram in an entity that performs a role in one given system. This could be a person, organization or an external system usually drawn like skeleton.

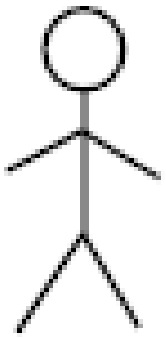


Fig1:Actor

Use case

A use case represents a function or an action within the system. Its drawn as an oval and named with the function.

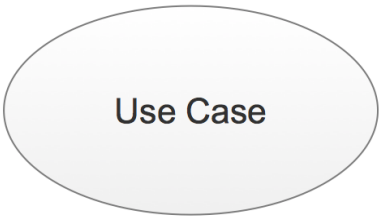


Fig2:Use Case

System

System is used to define the scope of the use case and drawn as a rectangle. This is an optional element but useful when your visualizing large systems. For example you

can create all the use cases and then use the system object to define the scope covered by your project. Or you can even use it to show the different areas covered in different releases.

Fig3 :system

Package

Package is another optional element that is extremely useful in complex diagrams.

Similar to use class diagrams, packages are used to group together use cases.

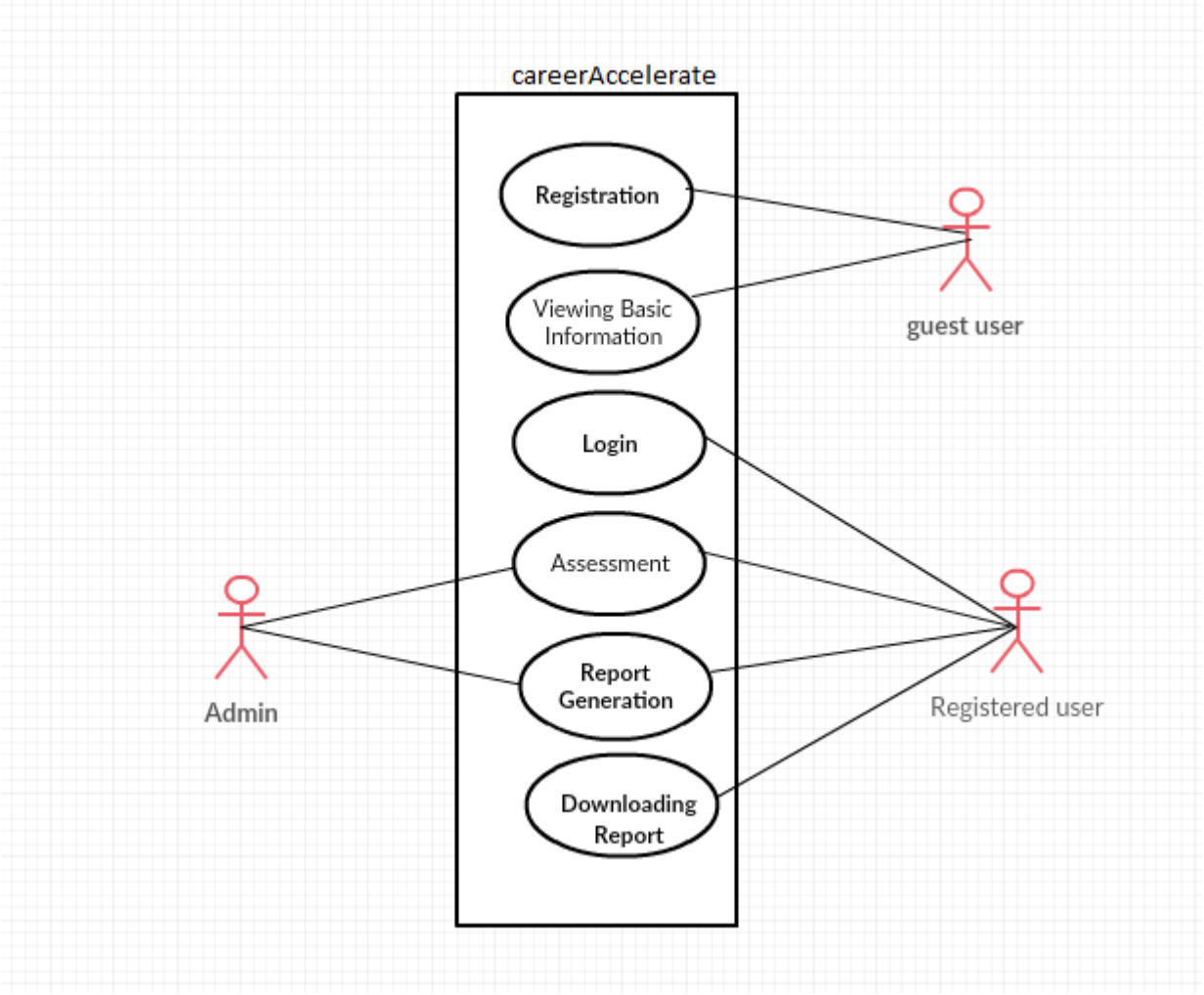


Fig 4: usecase diagram

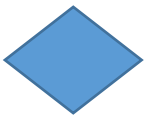
3.2.2 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores Activity diagrams are constructed from a limited number of shapes, connected with arrows.

The most important shape types:

- *ellipses* represent *actions*

- *diamonds* represent *decisions*



- *bars* represent the start (*split*) or end (*join*) of concurrent activities

- a *black circle* represents the start (*initial node*) of the workflow

- an *encircled black circle* represents the end (*final node*)

Arrows run from the start towards the end and represent the order in which activities happen.

Activity diagrams can be regarded as a form of a structured flowchart combined with a traditional data flow diagram. Typical flowchart techniques lack constructs for expressing concurrency.[5]However, the join and split symbols in activity diagrams only resolve this for simple cases; the meaning of the model is not clear when they are arbitrarily combined with decisions or loops.[*citation needed*]

UML activity diagrams in version 2.x can be used in various domains, e.g. in design of embedded systems. It is possible to verify such a specification using model checking technique.

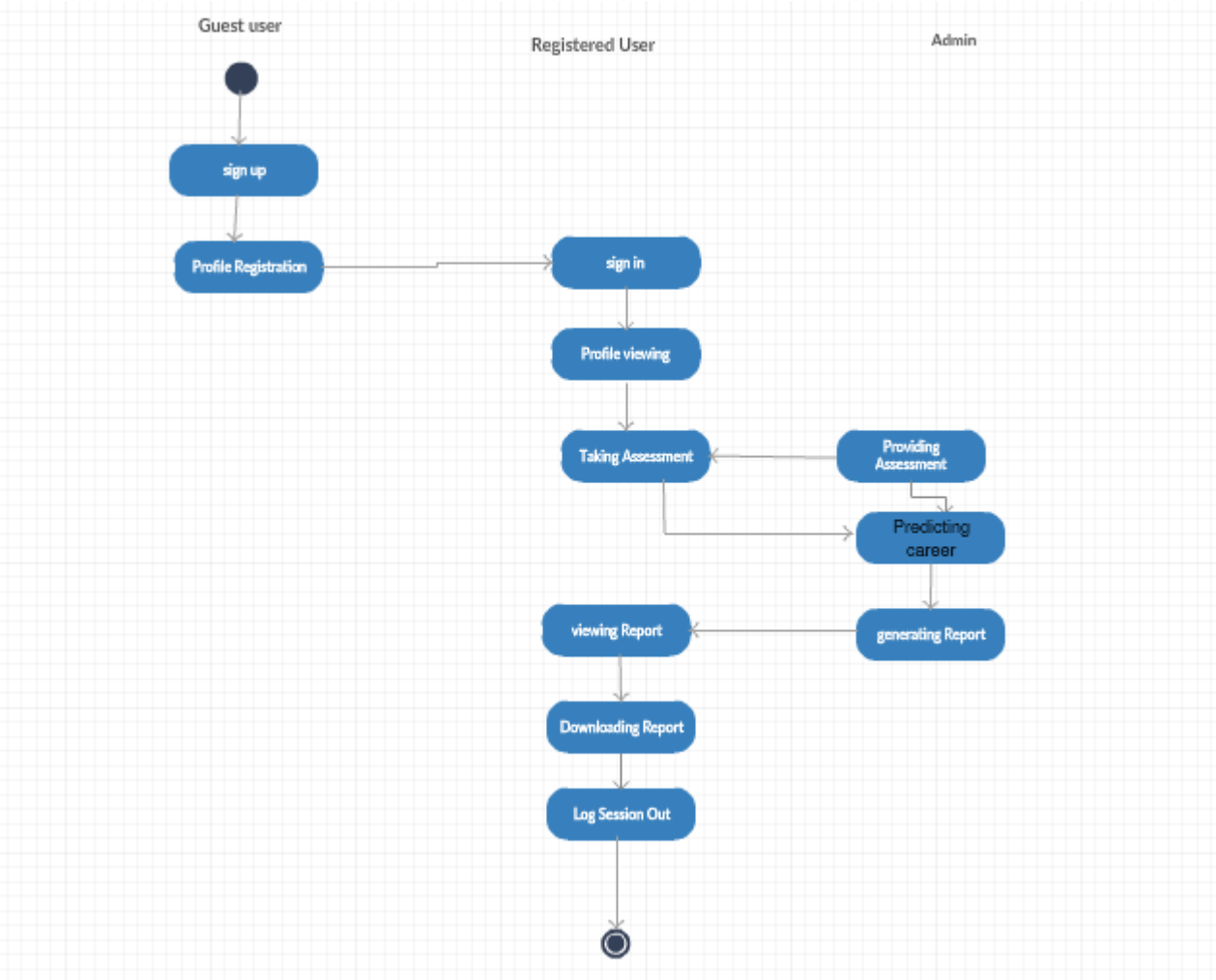


Fig 4

3.2 SYSTEM DESIGN

3.2.1 Input Design

The input is the set of values that is provided by the user to the system.The input design must enable the user to provide the error free input to the system for efficient processing.The data is fed into the system using simple interactive xml pages.The pages have been supplied with messages so that user can enter data without facing any difficulty.The data is validated wherever it requires in the project

The main objectives of the input design are as follows:

- Produce effective method of input
- Achieve high level accuracy
- Ensure that the input is acceptable and understood by the user

3.2.2 Output Design

The primary consideration in the design of all output is the information requirement and other objective of the users. It is the most important and direct source of information to the user. A major form of output is a hard copy. Print out should be designed around the output requirements of the user. Each output should be given a specific name or title. The output data is displayed on the visual display unit and output can be redirected to printers and or sorted in a file for later use.

3.2.3 Database Design

Database is a design to manage large bodies of information. The management of data involves both the definition of structures for the storage information. In addition, the database system must provide for the safety of the information solved, despite system crashes or due to attempts at unauthorized access. For developing an efficient database we have to fulfil certain condition such as controlled redundancy

- Defining the data
- Inputting the data
- Locating the data
- Accessing the data
- Communicating the data
- Revising the data

OBJECTIVES OF DATABASE

In the database design, several objectives are consid- ered such as

- Control of data Integrity
- Ease of use
- Control of redundancy
- Control of security
- Data independence(Logical and physical)
- Data storage protection(Record level and Table level)
- System performance
- System functions
- System compatibility

For achieving the above mentioned criterias we have to make use various features that are available with the RDBMS by enforcing integrity constraints, we can ensure data integrity and reduce data inconsistency to a great extent. Recovery from failures can overcome using backup facilities. By using table level as well as row level locking facilities , wecan avoid concurrent access normalize. Another important features of RDBMS is the logical and physical data independence. In addition to security mechanism provided by RDBMS, we have provided system password to near system

NORMALIZATION: Normalization is the term obtained from the Latin word NORMA which means that square used by the carpenter .Normalization is the process of simplifying the relationship between data elements in a record, through normalization a collection of data I a record structure is replaced by successive record structures that are simpler and can be managed efficiently. While designing the database, we have to implement the concept of normalization to avoid data redundancy in the database. Normalization is carried out for four reasons.

- To structure the data so that any pertinent relationship between entities can be represented.
- To permit simple retrieval of data in response to query and reports required.
- To simplify data maintenance procedures such as insertion, deletion and updating.
- To reduce the need to be structure or reorganize data with new application requirements arise.

The major normalization strategies are

- First Normal Form
- Second Normal Form
- Third Normal Form
- Boyce/Codd Normal Form(BCNF)

FIRST NORMAL FORM: First Normal Form is achieved when all repeating groups in a record are removed, so that record is of fixed length. A repeating group, reoccurrence of a data item or group of data item within a record indicates another relation.

SECOND NORMAL FORM: Second Normal Form is achieved when a record is in first normal form and each item in the record is functionally depend on the primary key for identification. In other words, analyst seeks functional dependency. A data item is functionally dependent of its value is uniquely associated with a specific data item is functionally dependent of its value is uniquely associated with a specific item. To achieve second normal form every column in a table that is not dependent on the primary key of the record should be removed and used to form a separate relation.

THIRD NORMAL FORM: Third Normal Form is achieved when all transitive dependencies are removed from a record. That is, if A is functionally dependent on B and B is functionally dependent on C, then A is functionally dependent on C.

BOYCE/CODD NORMAL FORM(BCNF):

BCNF is often used to distinguish the new 3NF from the old. An attribute possible composite is called as determinant. If other attributes are fully functionally determined this attribute(or on which some other attribute is fully functionally dependent on this attribute).A table is in BCNF , if every determinant is a candidate key. To achieve a table is in BCNF, remove fields which are fully functionally dependent on a determinant, which is not act as a candidate key.

Table 1: Login Table

Field	**Constraints**	**Data Type**
Username	Primary key	Varchar2(15)
Password	Not null	Varchar2(20)
Email	Not null	Varchar2(30)

Table 2: Profile

Field	**Data Type**	**Constraints**
sId	Number(7)	Not null
Firstname	Varchar2(25)	Not null
Lastname	Varchar2(25)	Not null
Schoolname	Varchar2(25)	Not null
Stream	Varchar2(25)	Not null
Standard	Varchar2(25)	Not null
phonenumber	Varchar2(20)	Not null
Username	Varchar2(15)	Foreign Key

Exam

Field	**Data Type**	**constraints**
Username	Varchar2(15)	Foreign Key
Option1	Varchar2(15)	
Option2	Varchar2(15)	
Option3	Varchar2(15)	
Option4	Varchar2(15)	
Option5	Varchar2(15)	
Option6	Varchar2(15)	
Option7	Varchar2(15)	
Option8	Varchar2(15)	
Option9	Varchar2(15)	
Option10	Varchar2(15)	
Optio11	Varchar2(15)	
Option12	Varchar2(15)	
Option13	Varchar2(15)	
Option14	Varchar2(15)	
Option15	Varchar2(15)	
Var1	Varchar2(15)	
Var2	Varchar2(15)	
Var3	Varchar2(15)	
Var4	Varchar2(15)	
Var5	Varchar2(15)	

3.4 TOOLS AND APPLICATION

3.4.1 ANGULAR

UI designing is one of the most crucial parts of any application. UI is where the user interacts with any application, hence it has to be designed in a user-friendly manner. Many frameworks have come into existence for the same. Angular is one such very powerful framework for building client applications using any scripting language such as JS or TypeScript. The most preferred scripting language for Angular is TypeScript which can again be compiled to JavaScript. Any UI designer would appreciate Angular if he already knows any traditional approaches as this framework helps us in designing application with better performance and maintainability for both mobiles and desktops.

AngularJS is based on the model view controller, whereas Angular 2 is based on the components structure. Angular 4 works on the same structure as Angular2 but is faster when compared to Angular2.

Angular4 uses TypeScript 2.2 version whereas Angular 2 uses TypeScript version 1.8. This brings a lot of difference in the performance.

To install Angular 4, the Angular team came up with Angular CLI which eases the installation.

The Angular 4 app folder has the following **folder structure** −

- **e2e** − end to end test folder. Mainly e2e is used for integration testing and helps ensure the application works fine.
- **node_modules** − The npm package installed is node_modules. You can open the folder and see the packages available.
- **src** − This folder is where we will work on the project using Angular 4.

The Angular 4 app folder has the following **file structure** −

- **.angular-cli.json** − It basically holds the project name, version of cli, etc.
- **.editorconfig** − This is the config file for the editor.
- **.gitignore** − A .gitignore file should be committed into the repository, in order to share the ignore rules with any other users that clone the repository.
- **karma.conf.js** − This is used for unit testing via the protractor. All the information required for the project is provided in karma.conf.js file.
- **package.json** − The package.json file tells which libraries will be installed into node_modules when you run npm install.

The src folder is the main folder, which internally has a different file structure.

app-It contains the files described below. These files are installed by angular-cli by default.

- app.module.ts − If you open the file, you will see that the code has reference to different libraries, which are imported. Angular-cli has used these default libraries for the import – angular/core, platform-browser. The names itself explain the usage of the libraries.

They are imported and saved into variables such as **declarations, imports, providers**, and **bootstrap**.

declarations − In declarations, the reference to the components is stored. The Appcomponent is the default component that is created whenever a new project is initiated. We will learn about creating new components in a different section.

imports − This will have the modules imported as shown above. At present, BrowserModule is part of the imports which is imported from @angular/platform-browser.

providers − This will have reference to the services created. The service will be discussed in a subsequent chapter.

bootstrap − This has reference to the default component created, i.e., AppComponent.

- **app.component.css** − You can write your css structure over here. Right now, we have added the background color to the div as shown below.
- **app.component.html** − The html code will be available in this file.

This is the default html code currently available with the project creation.

- **app.component.spec.ts** − These are automatically generated files which contain unit tests for source component.
- **app.component.ts** − The class for the component is defined over here. You can do the processing of the html structure in the .ts file. The processing will include activities such as connecting to the database, interacting with other components, routing, services, etc.

Angular and TypeScript

Angular is one of the most powerful client-side U.I. framework which can be used to develop **complex**, **customizable**, **modern**, **responsive** and **userfriendly** web applications. Some such applications are PayPal, Netflix, Weather etc.Angular is a single framework which addresses concerns of both mobile and desktop application.The Angular framework has chosen a **new statically-typed**, **client-side scripting** language called TypeScript, which

beautifully works around most if not all such pitfalls of JavaScript. The Angular team recommends the usage of TypeScript for Angular applications. Hence, to start building Angular applications, we would need to learn to write simple TypeScript code first and then look at the Angular application design.

Pitfalls of JavaScript

JavaScript is the language used for client side scripting. We can do client side validations, DOM manipulation, Ajax calls etc using JavaScript. We can also use JavaScript frameworks for writing complex business logic which runs on the client side.

As the complexity of the JavaScript code increases, it gradually becomes difficult in coding and maintaining. This is because of the following pitfalls of JavaScript language.

Dynamic Typing: Dynamic typing means deciding the data type of the variable dynamically at runtime which results in recompilation every time the code is executed.

Interpreted Language: Interpreted Language is a language in which the code instructions are executed directly without prior compilation to machine-language instructions hence we will not get to know the errors until the code is executed.

Minimal Object Oriented support: JS supports minimal Object Oriented concepts like classes, encapsulation, inheritance which helps in readability and reusability of the code.

Minimal IDE support: Integrated development environment (IDE) is a software application that provides all necessary options like code refactoring, intellisense support, Debugging support to software programmers for software development which is least supported in JS.

Albeit all these shortcomings, we are still dependent on JS as it is the most common scripting language for browsers. But as programmers, we might be in luck.

The solution can be to choose a language which is rich in features and the code can be converted to JavaScript for browsers. This process of converting code written in one language into another language is generally called Transpilation.

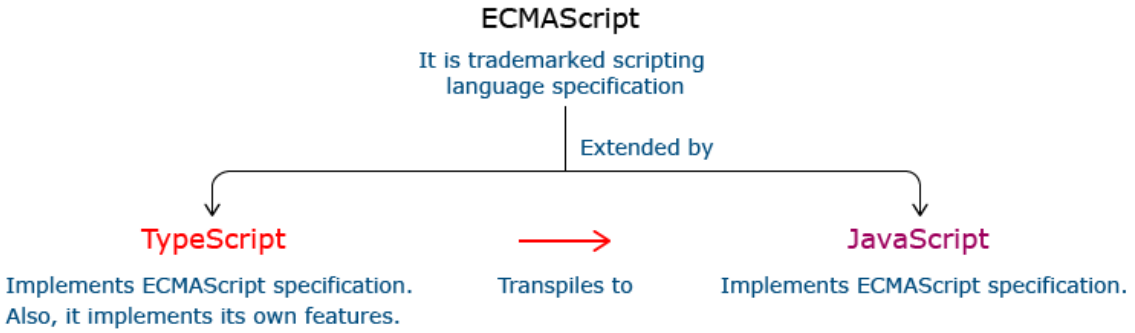
TypeScript is one such language whose code can be transpiled to JavaScript. This conversion is required because browser cannot understand TypeScript code.

What is TypeScript

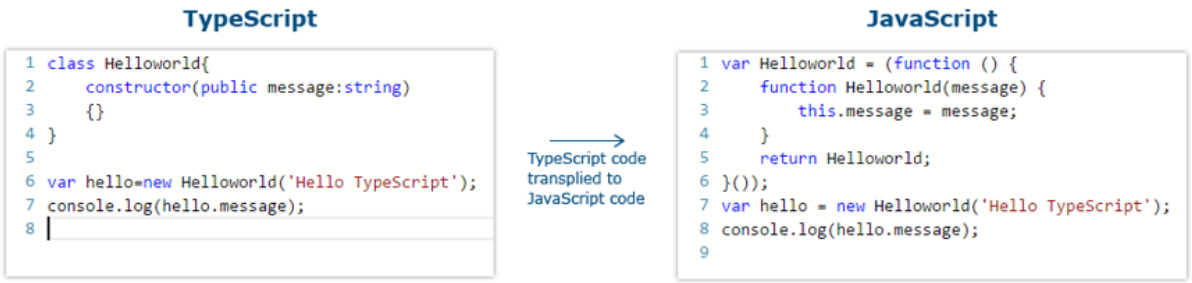
TypeScript is a typed superset of JavaScript that transpiles to JavaScript.

- TypeScript makes the development of JavaScript nearer to a more traditional object oriented experience.
- TypeScript is based on **ECMAScript** 7 proposals.

- Apart from the EcmaScript specification, TypeScript has its own features as well.
- Any valid JavaScript is TypeScript.



Relationship Between TypeScript and JavaScript



In the code given above, the TypeScript class HelloWorld is converted to a self invoking function in JavaScript when transpiled.

Features Of TypeScript

Static Typing: It adds static typing to JavaScript, due to which the readability of the code improves and also helps in finding more early compilation errors than the runtime errors.

Modules support: TypeScript provides an option to create modules so that we can modularize the code for easy maintenance. Modules also help in making the application scalable.

Object Oriented Programming: TypeScript supports object oriented programming features such as classes, encapsulation, interface, inheritance and so on which help in creating highly structured and reusable code.

Open Source: TypeScript is open source. The source code of TypeScript can be downloaded from github.

Cross Platform: It works across platforms.

Tooling Support: TypeScript works extremely well with Sublime Text, Eclipse, and almost all major IDEs as compared to JavaScript.

3.4.2 VISUAL STUDIO CODE IDE

Visual Studio Code is a source-code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging, embedded Git control and GitHub, syntax highlighting, intelligent code completion, snippets, and code refactoring. It is highly customizable, allowing users to change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. The source code is free and open source and released under the permissive MIT License. The compiled binaries are freeware and free for private or commercial use.

Visual Studio Code is based on Electron, a framework which is used to deploy Node.js applications for the desktop running on the Blink layout engine. Although it uses the Electron framework, the software does not use Atom and instead employs the same editor component (codenamed "Monaco") used in Azure DevOps (formerly called Visual Studio Online and Visual Studio Team Services).

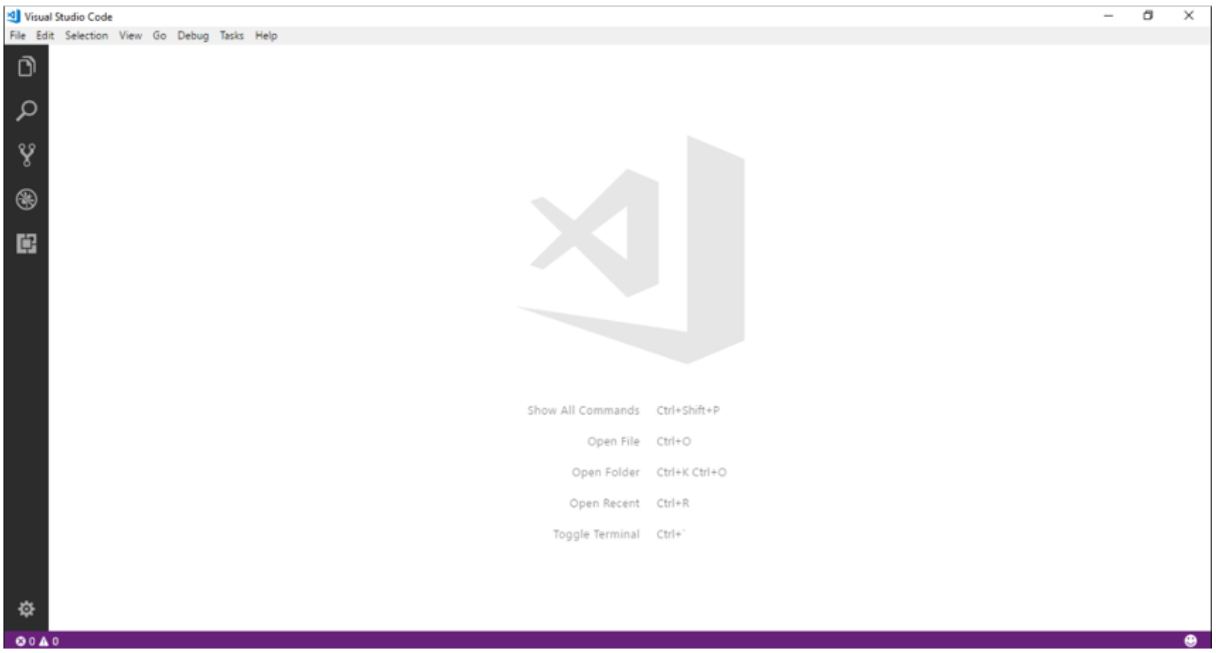
Typescript code can be written using several IDEs such as - Eclipse IDE, NetBeans IDE, Visual Studio Code IDE etc. We are prefer using Visual Studio Code IDE as it is a lightweight cross-platform editor which comes with built-in TypeScript support.

Typescript code is written in Visual Studio Code IDE in the following way:

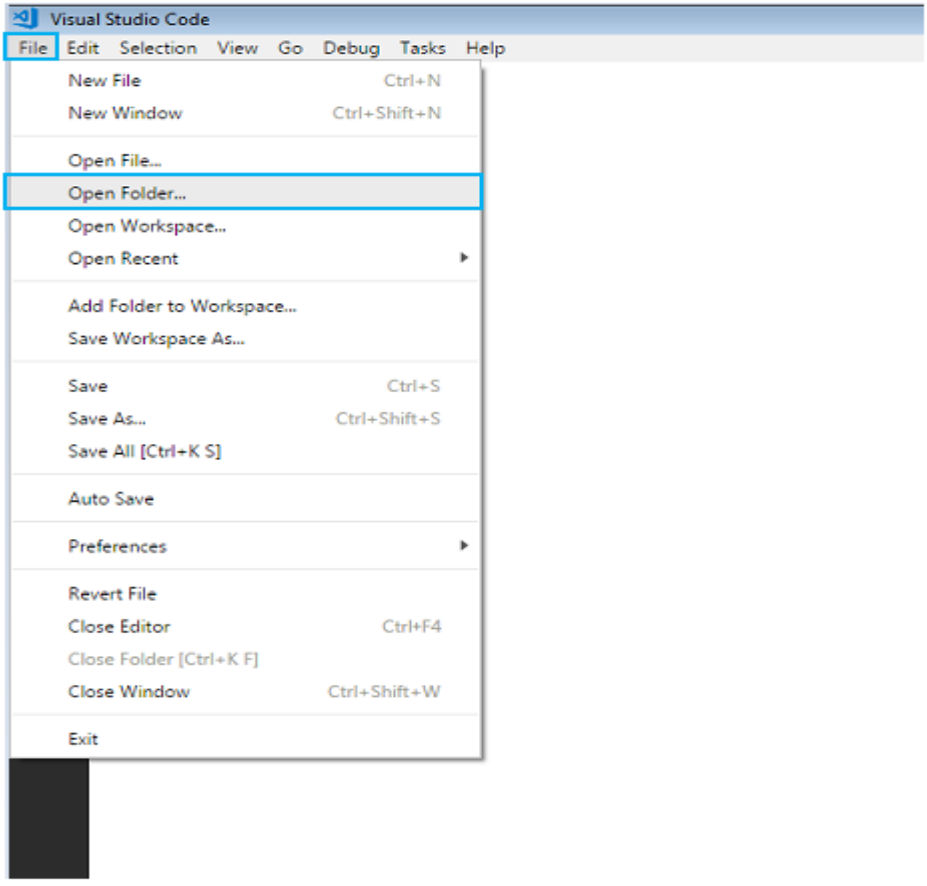
Step 1: Create a folder on your desktop. This folder will be used as a work-space for Visual Studio Code. Let us name it **TypeScript_WorkSpace.**

Step 2: Create a folder and name it as **Demo**, inside the previously created folder. This folder will be used as a project folder.

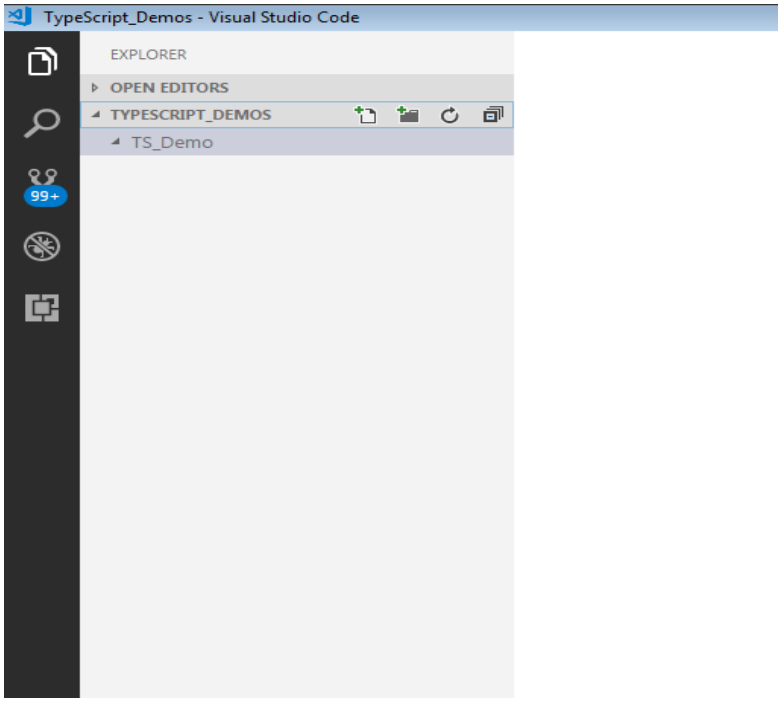
Step 3: Launch Visual Studio Code IDE. Once launched, close the welcome page. We should get a screen similar to the one shown below.



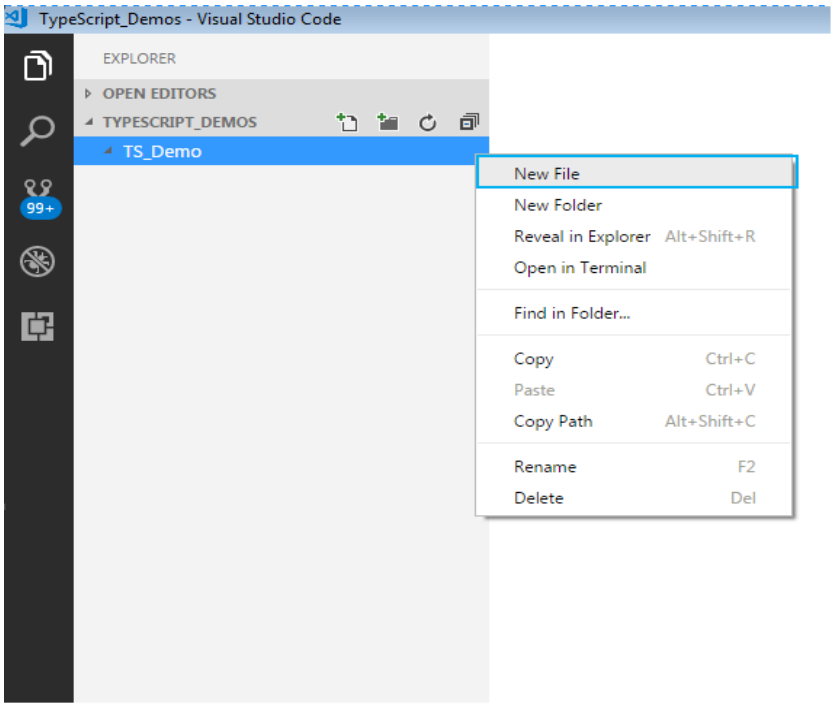
Step 4: Next step is to select a workspace. From the **File** menu, select **OpenFolder,** browse for the TypeScript_Demos folder and select it.

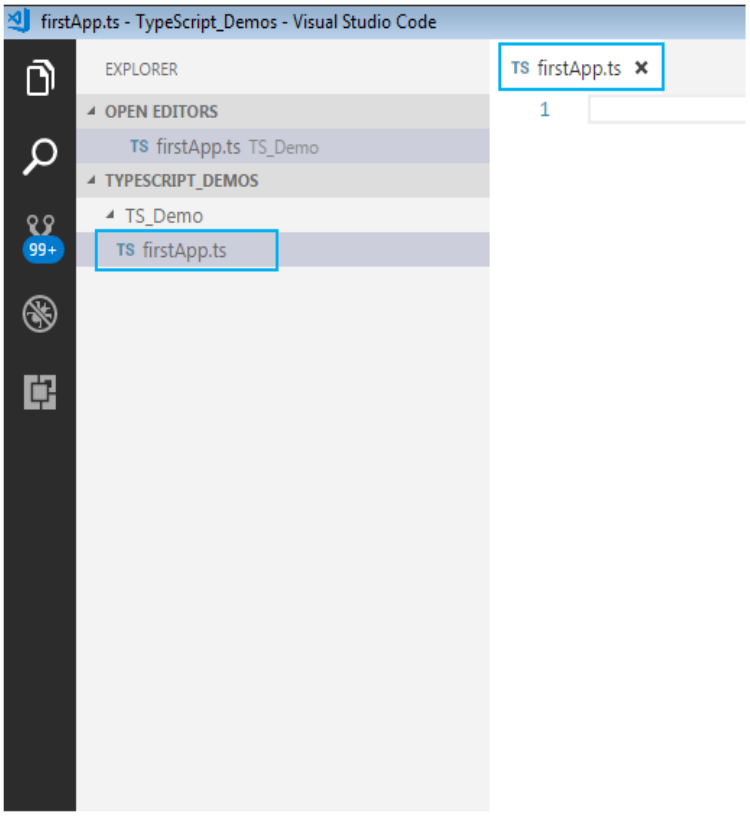


Step 5: The IDE will restart and the folder will be selected as work-space. We will also have the **TS_Demo** project folder as shown below.



Step 6: Let us create a typescript file. Create a file named **firstApp.ts** in the project folder.





To start with the first application in TypeScript, in already created firstApp.ts file, give a console.log statement and save it.

Here is how to do this:

Step 7: We shall print a welcome message when we run the above file. To do so, let us code as shown

console.log("Hello! Welcome to TypeScript");

Step 8:

To execute the code, we will have to navigate to the project folder. For this, we will use windows command prompt through VSCode. Open the command prompt by following step:

- Navigate to View tab in VSCode.
- Select Integrated Terminal.

Alternatively, you can use the shortcut, **ctrl+`**, to open the command prompt. We will get the command prompt as shown below.

From the command prompt navigate to the folder in which the ts file resides and **transpile** the **.ts** file using the **tsc**command as follows:

```
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Users\likhith.kolayari\Desktop\Demo> D:
PS D:\> cd TypeScript-WorkSpace
PS D:\TypeScript-WorkSpace> tsc firstApp.ts
PS D:\TypeScript-WorkSpace>
```

Step 9: Run the transpiled **.js** file using **node** command . On execution of the code, the output is displayed on the console as shown.

```
PS D:\TypeScript-WorkSpace> tsc firstApp.ts
PS D:\TypeScript-WorkSpace> node firstApp.js
Hello! Welcome to TypeScript
PS D:\TypeScript-WorkSpace>
```

3.4.2 ORACLE

The Oracle Server is a relational database management system that provides an open, comprehensive, and integrated approach to information management. An Oracle Server consists of an Oracle database and an Oracle instance. The following sections describe the relationship between the database and the instance.

Structured Query Language (SQL)

SQL (pronounced SEQUEL) is the programming language that defines and manipulates the database. SQL databases are relational databases; this means simply that data is stored in a set of simple relations. A database can have one or more tables. And each table has columns and rows. A table that has an employee database, for example, might have a column called employee number and each row in that column would be an employee's employee number.

You can define and manipulate data in a table with SQL commands. You use data definition language (DDL) commands to set up the data. DDL commands include commands to creating and altering databases and tables.

You can update, delete, or retrieve data in a table with data manipulation commands (DML). DML commands include commands to alter and fetch dat. The most common SQL command is the SELECT command, which allows you to retrieve data from the

database.

In addition to SQL commands, the Oracle Server has a procedural language called PL/SQL. PL/SQL enables the programmer to program SQL statements. It allows you to control the flow of a SQL program, to use variables, and to write error-handling procedures.

Database Structure

An Oracle database has both a physical and a logical structure. Because the physical and logical server structure are separate, the physical storage of data can be managed without affecting the access to logical storage structures.

Physical Database Structure An Oracle database's physical structure is determined by the operating system files that constitute the database. Each Oracle database is made of three types of files: one or more datafiles, two or more redo log files, and one or more control files. The files of an Oracle database provide the actual physical storage for database information.

Logical Database Structure An Oracle database's logical structure is determined by

- one or more tablespaces. (A tablespace is a logical area of storage explained later in this chapter.)

- the database's schema objects. A *schema* is a collection of objects. *Schema objects* are the logical structures that directly refer to the database's data. Schema objects include such structures as tables, views, sequences, stored procedures, synonyms, indexes, clusters, and database links.

The logical storage structures, including tablespaces, segments, and extents, dictate how the physical space of a database is used. The schema objects and the relationships among them form the relational design of a database.

An Oracle Instance

Every time a database is started, a system global area (SGA) is allocated and Oracle background processes are started. The system global area is a an area of memory used for database information shared by the database users. The combination of the background processes and memory buffers is called an Oracle *instance.*

An Oracle instance has two types of processes: user processes and Oracle processes. A user process executes the code of an application program (such as an Oracle Forms

application) or an Oracle Tool (such as Server Manager).
Oracle processes are server processes that perform work for user processes and background processes that perform maintenance work for the Oracle Server.

3.4.3 Eclipse Luna

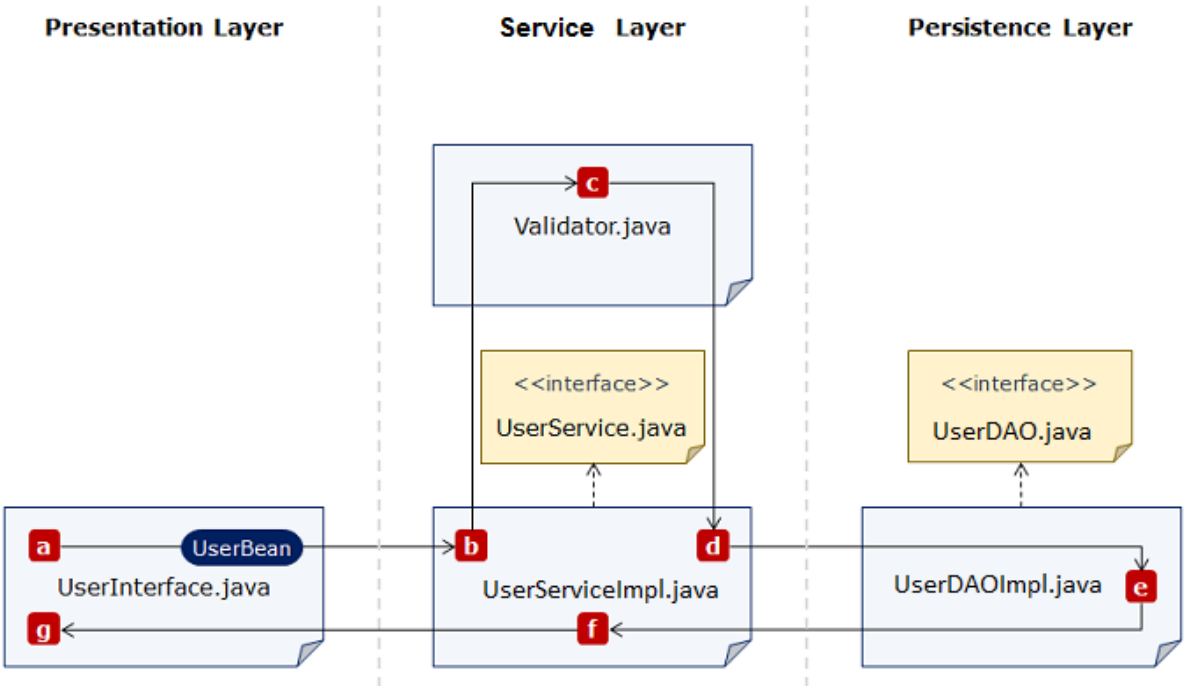
Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming languages via
plugins,including Ada, ABAP, C, C++, C#, Clojure, COBOL, D, Erlang, Fortran, G roovy, Haskell, JavaScript, Julia,[7] Lasso, Lua, NATURAL, Perl, PHP, Prolog, Pyt hon, R, Ruby (including Ruby on Rails framework), Rust, Scala, and Scheme. It can also be used to develop documents with LaTeX (via a TeXlipse plug-in) and packages for the software Mathematica. Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++, and Eclipse PDT for PHP, among others.

The Java Development Tools (JDT) project provides a plug-in that allows Eclipse to be used as a Java IDE, PyDev is a plugin that allows Eclipse to be used as a Python IDE, C/C++ Development Tools (CDT) is a plug-in that allows Eclipse to be used for developing application using C/C++, the Eclipse Scala plug-in allows Eclipse to be used an IDE to develop Scala applications and PHPeclipse is a plug-in to eclipse that provides complete development tool for PHP.

An eclipse perspective is the name given to an initial collection and arrangement of views and an editor area. The default perspective is called java. An eclipse window can have multiple perspectives open in it but only one perspective is active at any point of time. A user can switch between open perspectives or open a new perspective. The active perspective controls what appears in some menus and tool bars.
For implementing a user story we will create following classes:

- **UserService** – The business logic of this user story is implemented in this class. It interacts with the UserInterface in the presentation layer and UserDAO class in the persistence layer.
- **UserDAO** – This class interacts with the database and UserService class of service layer.
- **UserInterface** – This class acts as the user interface of the application and sends data to UserService class of service layer.
- **Validator -** This class validates the user data for correctness and interacts with UserService.
- **User** – This is a model class which is used to transfer data across different layers. It is a Java bean class having private properties and public getter and setter methods.

The following diagram depicts the classes present in each layer and their interactions along with the flow of data across the layers, for the success flow in the above user story:



1. **a-b:** The user bean is passed to UserServiceImpl
2. **c:** The bean is validated
3. **d-e:** The bean is passed to UserDAOImpl
4. **e:** DAO operations
5. **f:** Confirmation
6. **g:** Success message

CHAPTER 4
SYSTEM TESTING

Testing is the process of examining the software to compare the actual behavior with that of the excepted behavior. The major goal of software testing is to demonstrate that faults are not present. In order to achieve this goal the tester executes the program with the intent of finding errors. Though testing cannot show absence of errors but by not showing their presence it is considered that these are not present.

System testing is the first Stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operations commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct and the goal will be successfully achieved. A series of testing are performed for the proposed system before the proposed system is ready for user acceptance testing.

The purpose of system testing is to identify and correct errors in the candidate system. Testing is an important element of the software quality assurance and represents the ultimate review of specification, design and coding. The increasing visibility of the software as a system element and the costs associated with a software failure are motivated forces for well planned, through testing. Software testing is a critical element of software quality assurance and represents the ultimate quality review of specifications, design and code generation.

Once the source code has been generated, the program should be executed before the customer gets it with the specific intend of fining and removing all errors, test must be designed using disciplined techniques. Testing techniques provides the systematic guidance for designing tests. To uncover the errors in the program behavior function and performance the following steps to be done:

- Execute the integral logic of the software components.
- Execute the input and output domains of the program to uncover errors

During testing the system is used experimentally to ensure that the software does not fail, i.e., it will run according to the specification and in the way the user expects. Preparation of test data plays n vital role in the system testing. Different set of test data are generated and the system under study is tested using that data. While testing using test data errors are again uncovered and corrected using different testing techniques. System testing was conducted in order to detect errors and for comparing then the final system with the requirement specification report. That is, whether the system meets requirements. During testing the software was executed with a set of test cases and the output of the program for the test cases was evaluated to determine if the program is performing as it was expected to.

Testing presents, an interesting challenge for the software engineer attends to hold software from an abstract concept to an acceptable implementation. In testing engineer creates a series of test cases that occurs when errors are uncovered. Testing is the process of executing a program for finding errors. A good test is one that has high probability of finding an uncovered error. The turn error is used to refer the difference between the actual output of the software and the correct output. Fault is a condition that causes the software to fail to perform its required function. Software reliability is defined as the required function.

Software reliability is defined as the probability that the software will not undergo failure for a specified time under specified condition. Failure is the inability of a system or a component to perform a required function according to its specification. Different levels of testing were employed for software to make an error free, fault free and reliable. Basically in software testing four type of testing methods are adopted.

4.1 LEVELS OF TESTING

- Unit Testing
- Integration Testing
- System Testing
- User acceptance Testing

4.1.1 UNIT TESTING

Here we test each module individually and integrated the overall system. Unit testing focuses verification efforts even in the smallest unit of software design in each module. This is known as ”module testing”. The modules of this project are tested separately. This testing is carried out in the programming style itself. In this testing each module is focused to work satisfactorily as regard to expected output from the module. There are some validation checks for the fields. Unit testing gives stress on the modules of the project independently of one another, to find errors. Different modules are tested against the specifications produced during the design of the modules. Unit testing is done to test the working of individual modules with test servers. Program unit is usually small enough that the programmer who developed it can test it in a great detail. Unit testing focuses first on that the modules to locate errors. These error are verified and corrected and so that the unit perfectly fits to the project.

4.1.2 INTEGRATION TESTING

Data can be lost across an interface, one module can have an adverse effect on the other sub-functions, when combined they may not perform the desired functions. Integrated testing is the systematic testing to uncover the errors within the interface. This testing is done with simple data and the developed system has run successfully with this simple data. The need for integrated system is to find the overall system performance. The Modules of this project are connected and tested. After splitting the programs into units, the units were tested together to see the defects between each module and function. It is testing to one or more modules or functions together with the intent of finding interface defects between the modules or functions. Testing completed at as part of unit

or functional testing, integration testing can involve putting together of groups of modules and functions with the goal of completing and verifying meets the system requirements.

4.2 SYSTEM TESTING

System testing focuses on testing the system as a whole. System Testing is a crucial step in Quality Management Process.In the Software Development Life Cycle System Testing is the first level where the System is tested as a whole. The System is tested to verify whether it meets the functional and technical requirements. The application/System is tested in an environment that closely resembles the production environment where the application will be finally deployed. The perquisites for System Testing are: -

- All the components should have been successfully Unit Tested.
- All the components should have been successfully integrated.

Testing should be completed in an environment closely resembling the production en- vironment. When necessary iterations of System Testing are done in multiple environ- ments.

4.3 USER ACCEPTANCE TESTING

The system was tested by a small client community to see if the program met the re- quirements defined the analysis stage. It was fond to be satisfactory. In this phase, the system is fully tested by the client community against the requirements defined in the analysis and design stages, corrections are made as required, and the production system is built. User acceptance of the system is key factor for success of the system.

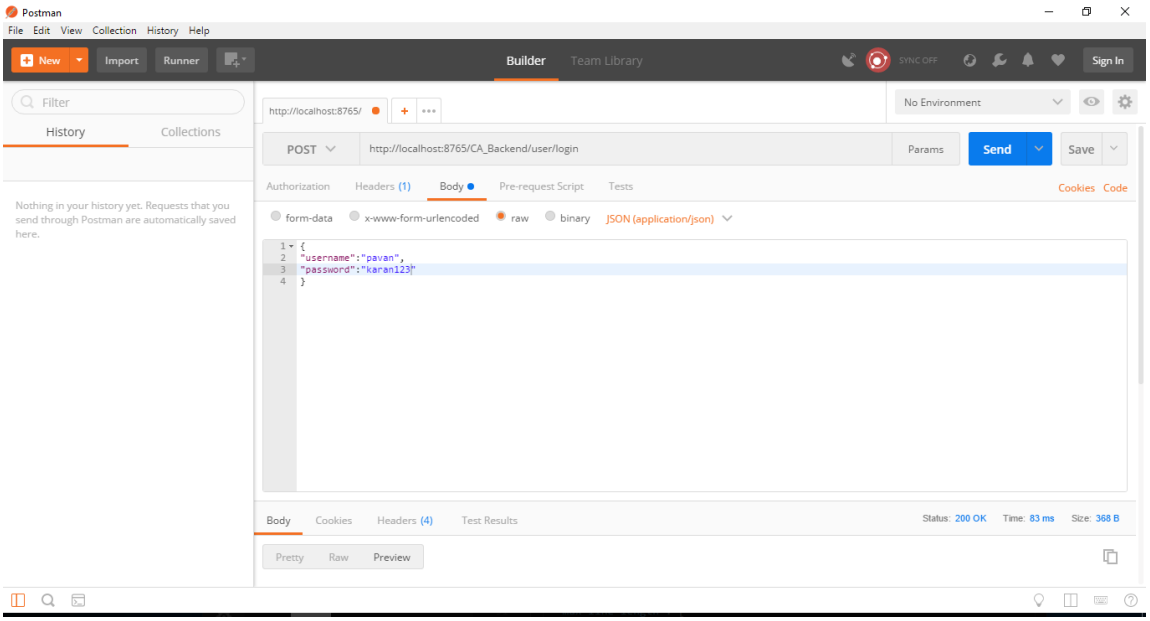


Figure 4.1: Test case using POSTMAN

CHAPTER 5
SYSTEM IMPLEMENTATION

The implementation is one phase of software development. Implementation is that stage in the project where theoretical design is turned into working system. Implementation involves placing the complete and tested software system into actual work environment. Implementation is concerned with translating design specification with source code. The primary goal of implementation is to write the source code to its specification that can be achieved by making the source code clear and straight forward as possible. Implementation means the process of converting a new or revised system design into operational one. Depending on the size of the organization and its requirements the implementation is divided into three parts

- Stage Implementation

  Here system is implemented in stages. The whole system is not implemented at once. Once the user starts working with system and is familiar with it, then a stage is introduced and implemented. Also the system is usually updated, regularly until a final system is sealed.

- Direct Implementation

  The proposed new system is implemented directly and the user starts working on the new System. The shortcoming, if any, faced are then rectified later.

- Parallel Implementation

  The old and the new system are not used simultaneously. This helps in comparison of the results from the two systems. Once the user is satisfied and his intended objectives are achieved by the new system, he stop using the old one. My project was implemented on approach of prototype model whose functionality was increased day by day, as the client was given full liberty in choosing his needs and gets to the maximum benefit out of the system developed.

Implementation is that process plan where the theoretical design is put into real test. All the theoretical and practical works are now implemented as a working system. This is most crucial stage in the life cycle of a project. The project may be accepted or rejected

depending on how it gathers confidence among the users. The implementation stage involves the following tasks

5.1 IMPLEMENTATION METHODS

Implementation of software refers to final installation of package in the real environment, to the satisfaction of the intended users and the successful operation of the system. Implementation is the stage of the project where the theoretical design is turned into a working system. Implementation includes all those activities that takes place to convert from the old system to new one. Proper implementation is essential to provide a reliable system to meet the organizational requirements.

5.2 IMPLEMENTATION PLAN

The Implementation Plan describes how the information system will be deployed, installed and transitioned into an operational system. The plan contains an overview of the system, a brief description of the major tasks involved in the implementation, the overall resources needed to support the implementation effort, and any site-specific implementation requirements. The plan is developed during the Design Phase and is updated during the Development Phase the finalversion is provided in the Integration and Test Phase and is used for guidance during the Implementation Phase.

Chapter 6:
CONCLUSION

The Project titled **"CareerAccelerate"** is a web based application done using Angular 4 as the front-end and oracle SQL+ as the back-end. CareerAccelerate is a venture aimed at providing career counselling services. It offers guidance to a variety of individuals seeking career help at different points in their career. CareerAccelerate is planning to move its business onto the digital platform In this project, a OCEAN model is used to predict the career of the user

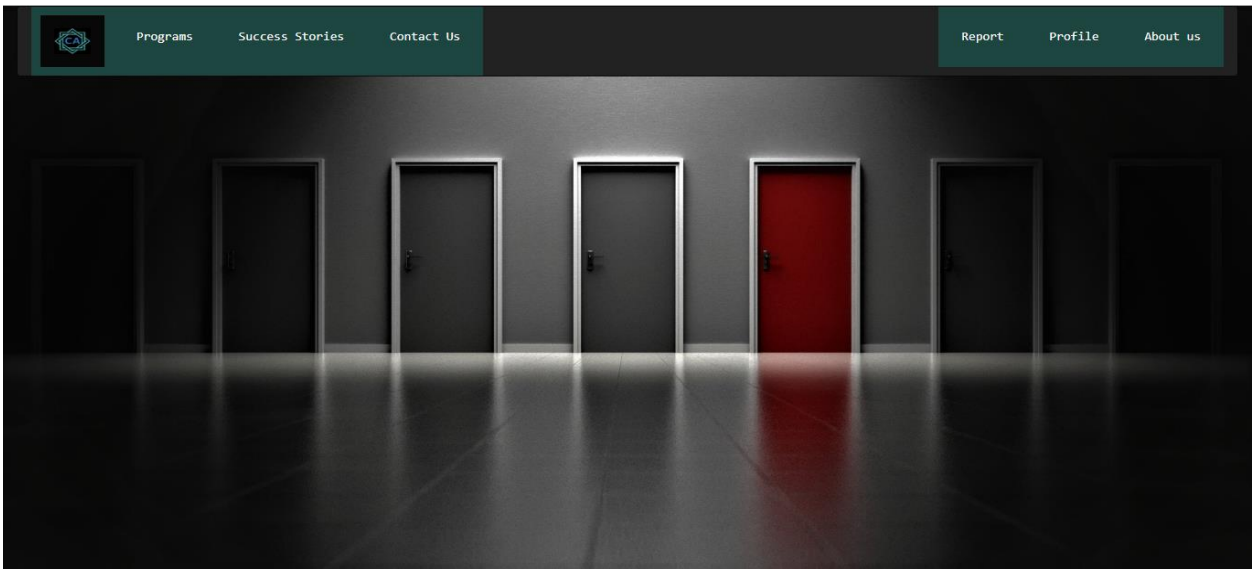
CHAPTER 7
REFERNCES

- HTML5 & CSS3: http://www.w3schools.com/
- jQuery: http://jquery.com/
- Bipin .C.Desai,An Introduction To Database System published by West.Publishing Company in 1990.
- Somer Vilie, Software Engineering published by Pearson in March13,2010.
- Sudarshan Silberschatz, Database System Concepts published by MCGraw- Hill Publication 2013.
- ASP.NET: http://www.w3schools.com/aspnet

Chapter 8

APPENDIX

SCREENSHOTS



8.1 front page

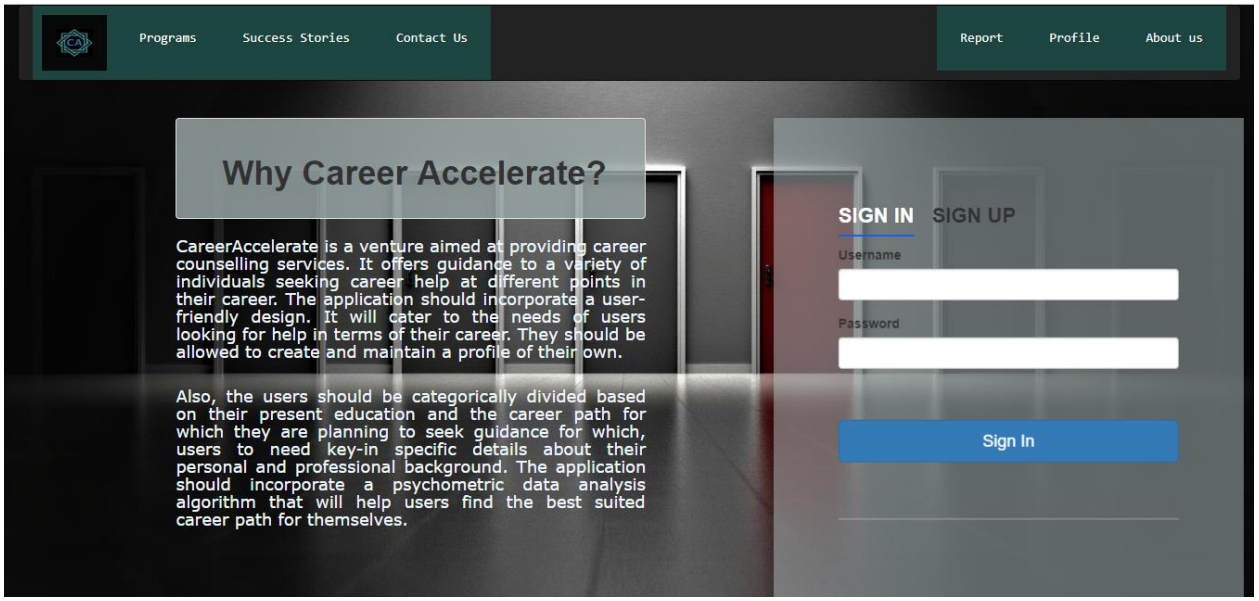


Fig 8.2 Main page

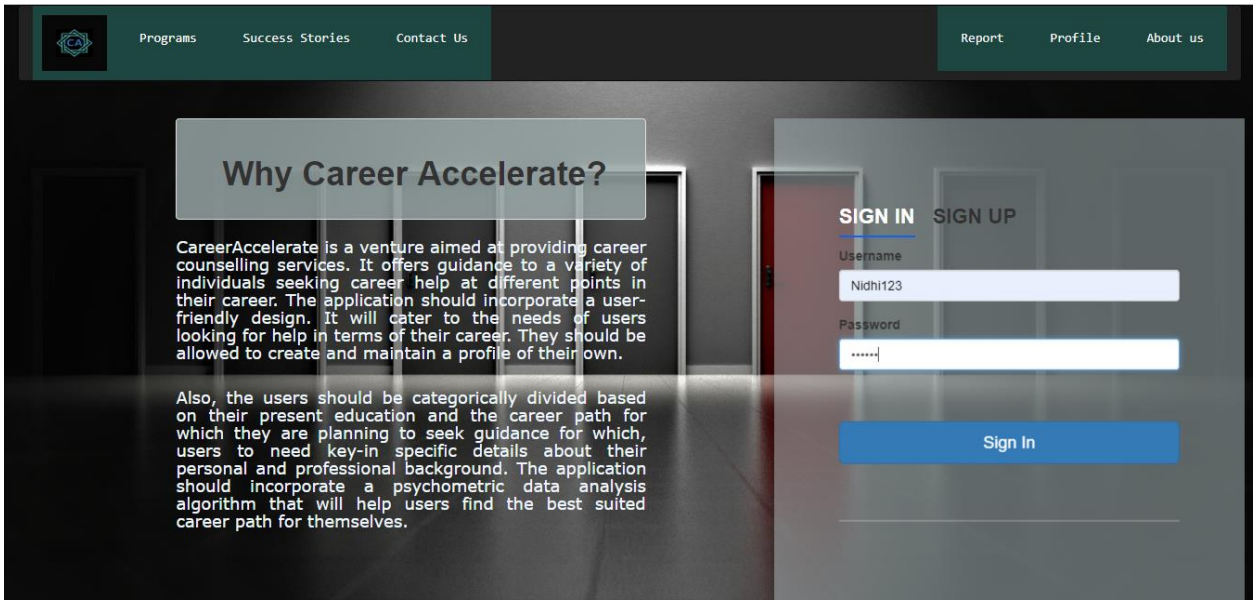


Fig 8.3 sign-in page

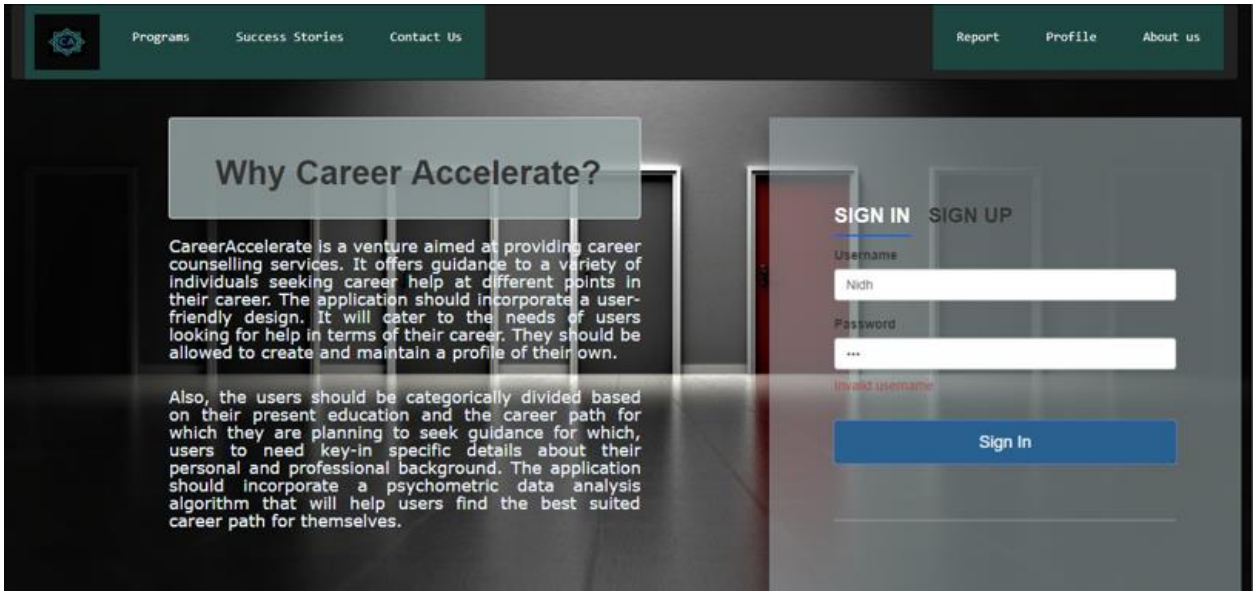
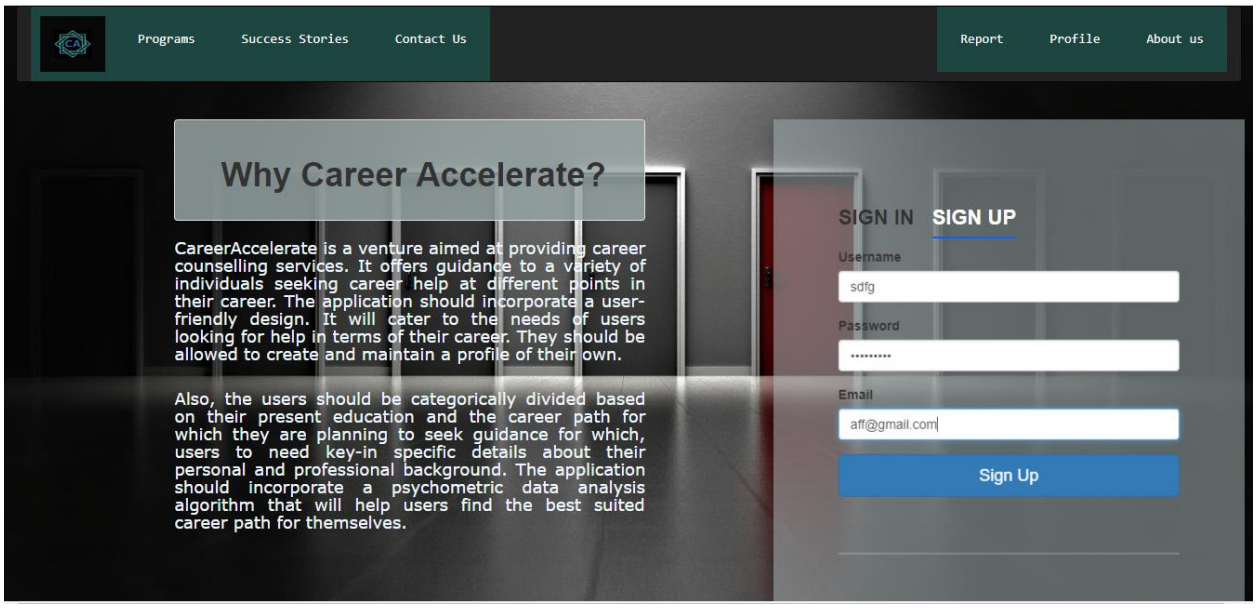
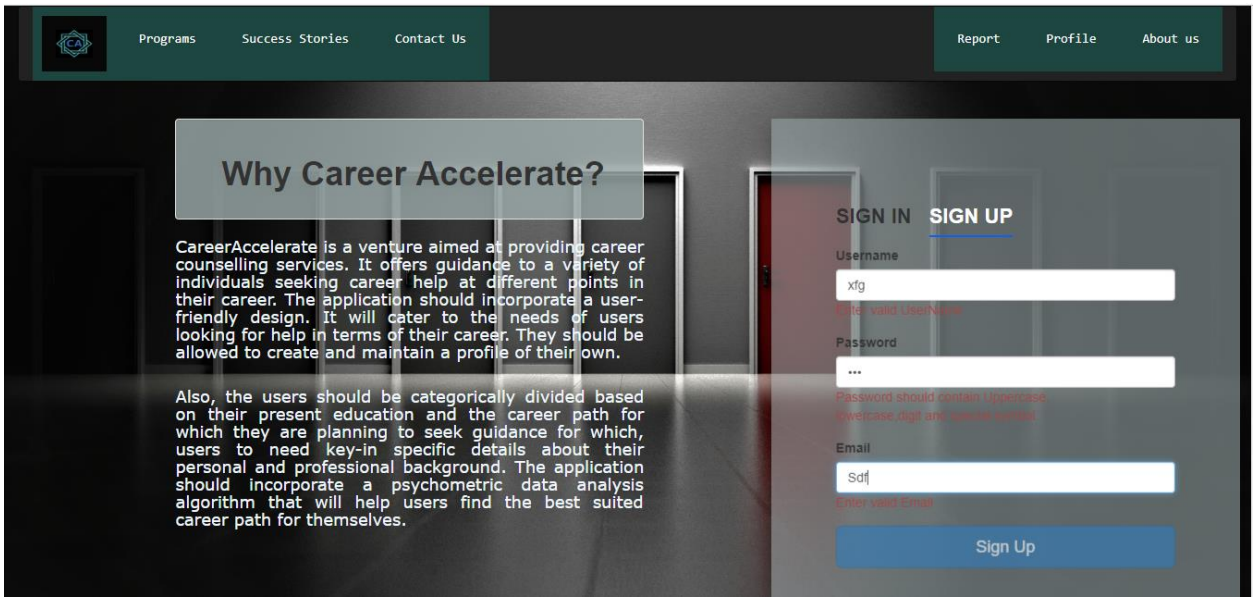


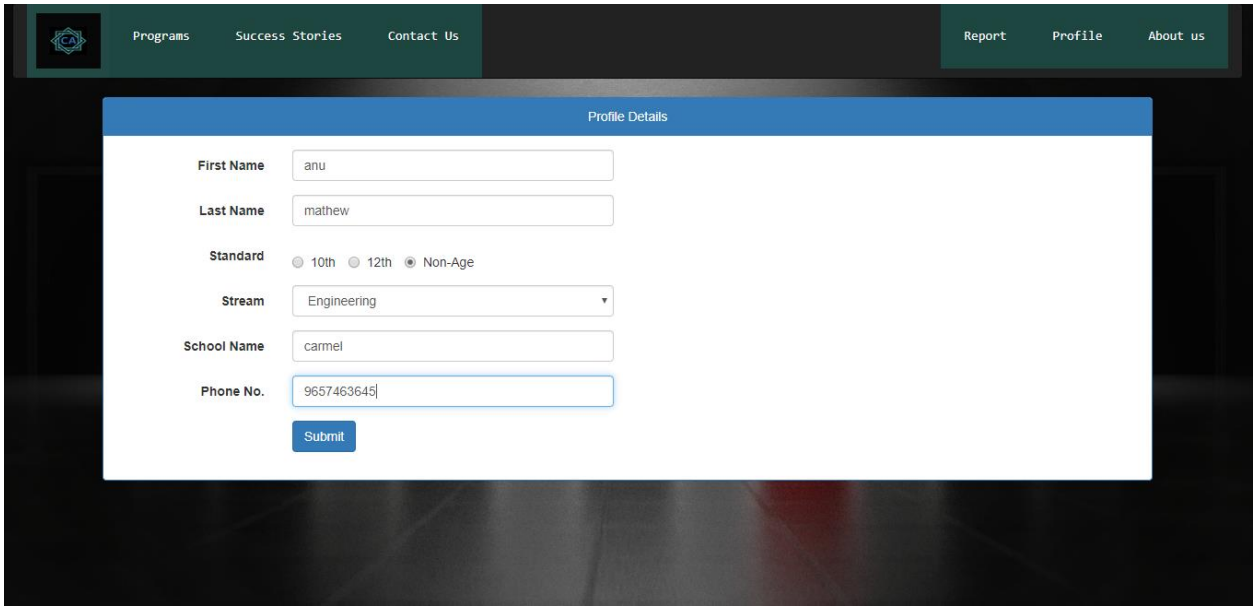
Fig 8.4 Invalid sign in



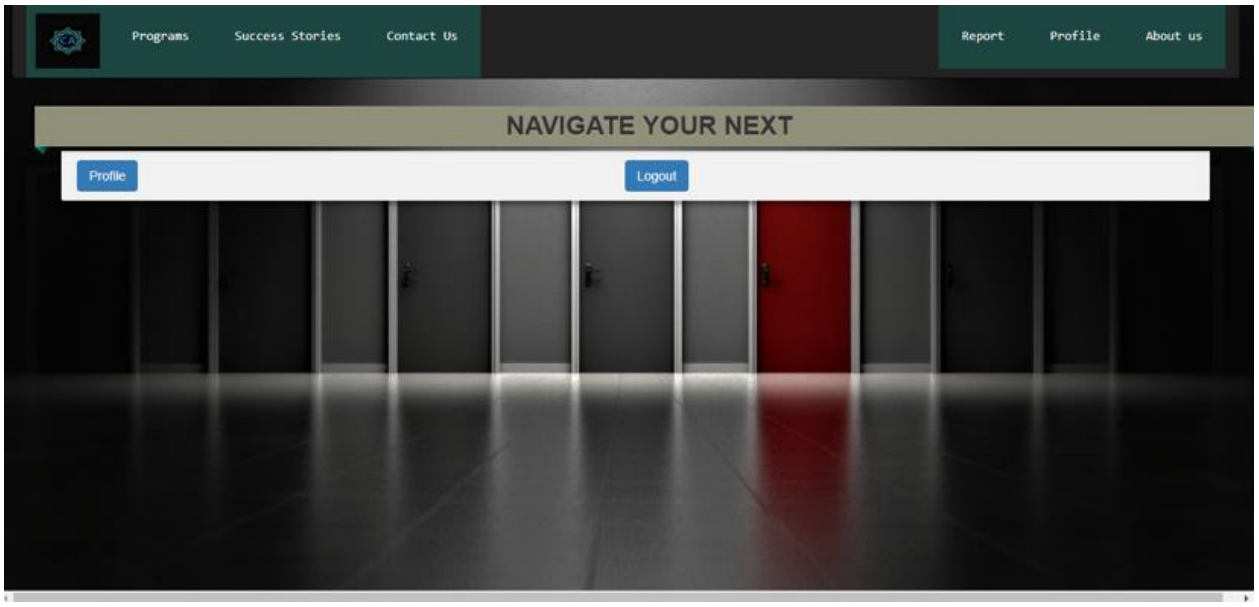
8.5 sign up page



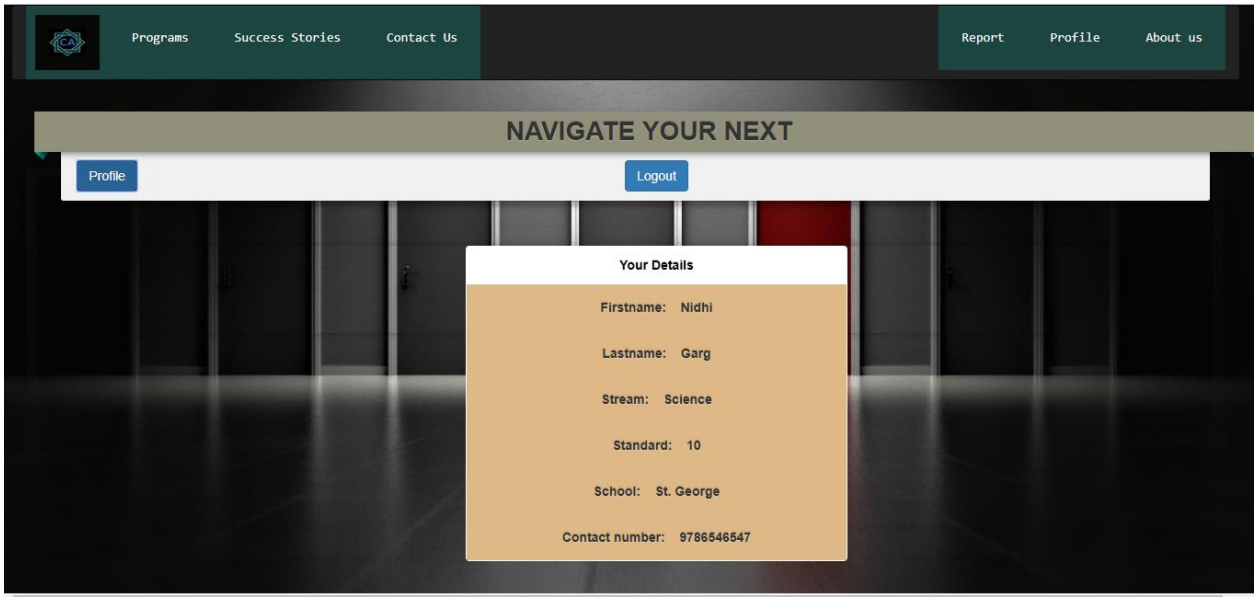
8.6 invalid sign up



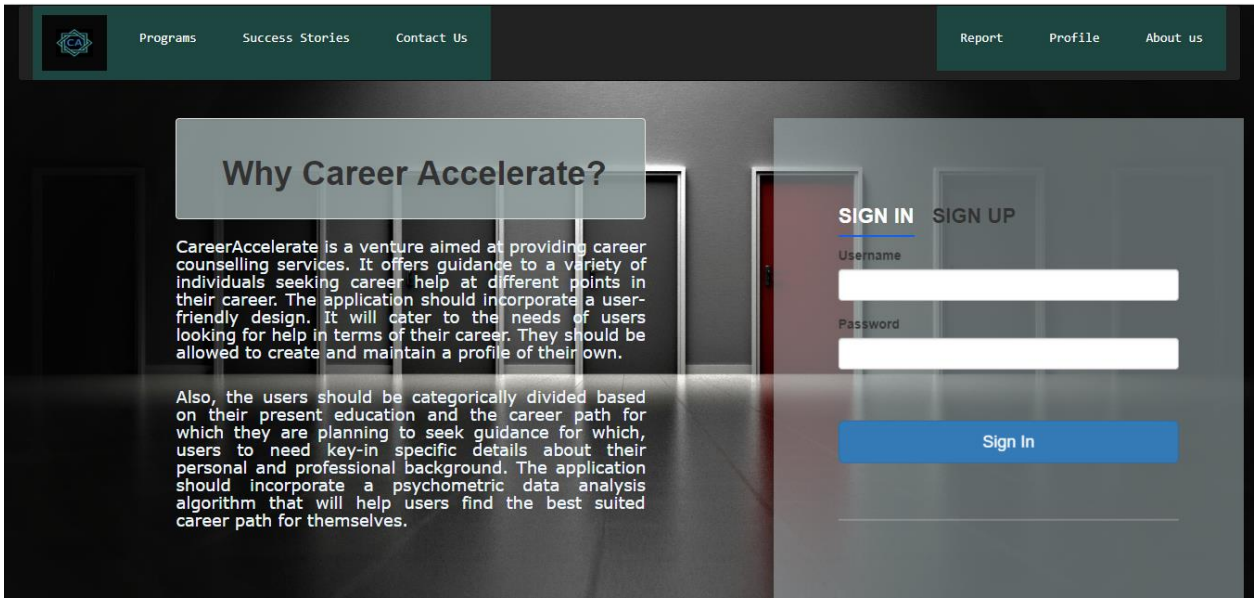
8.7 profile page



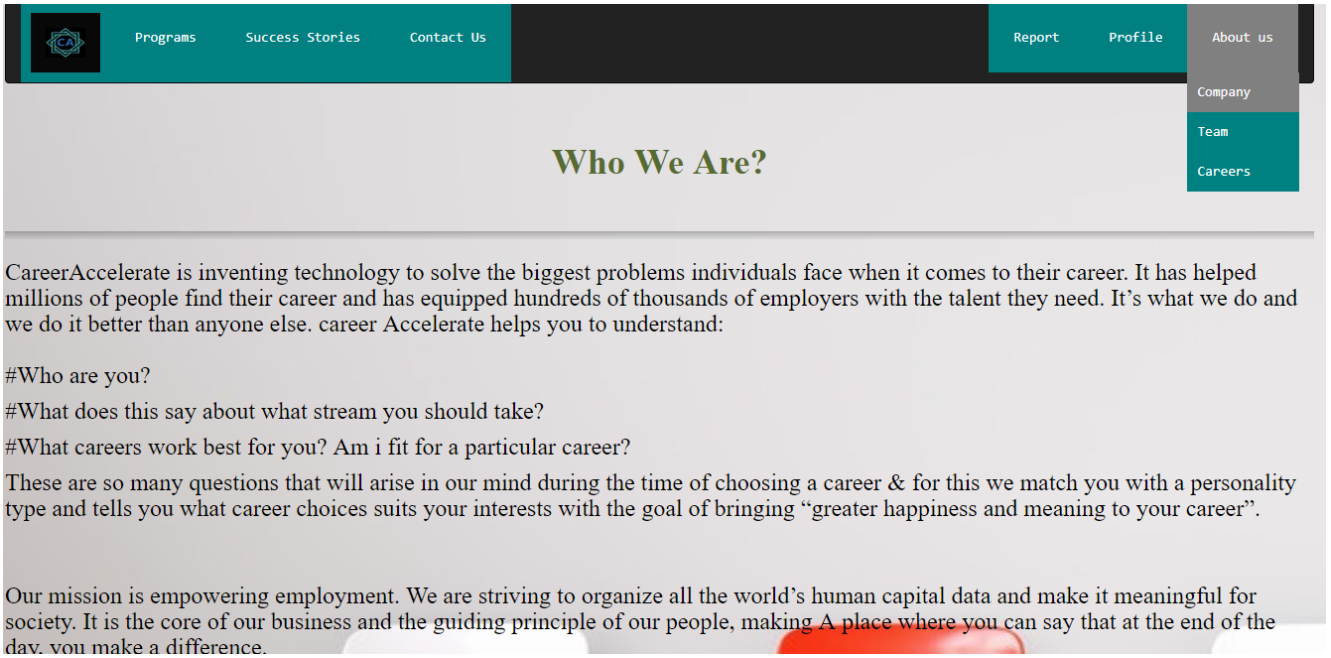
8.8 Profile main page



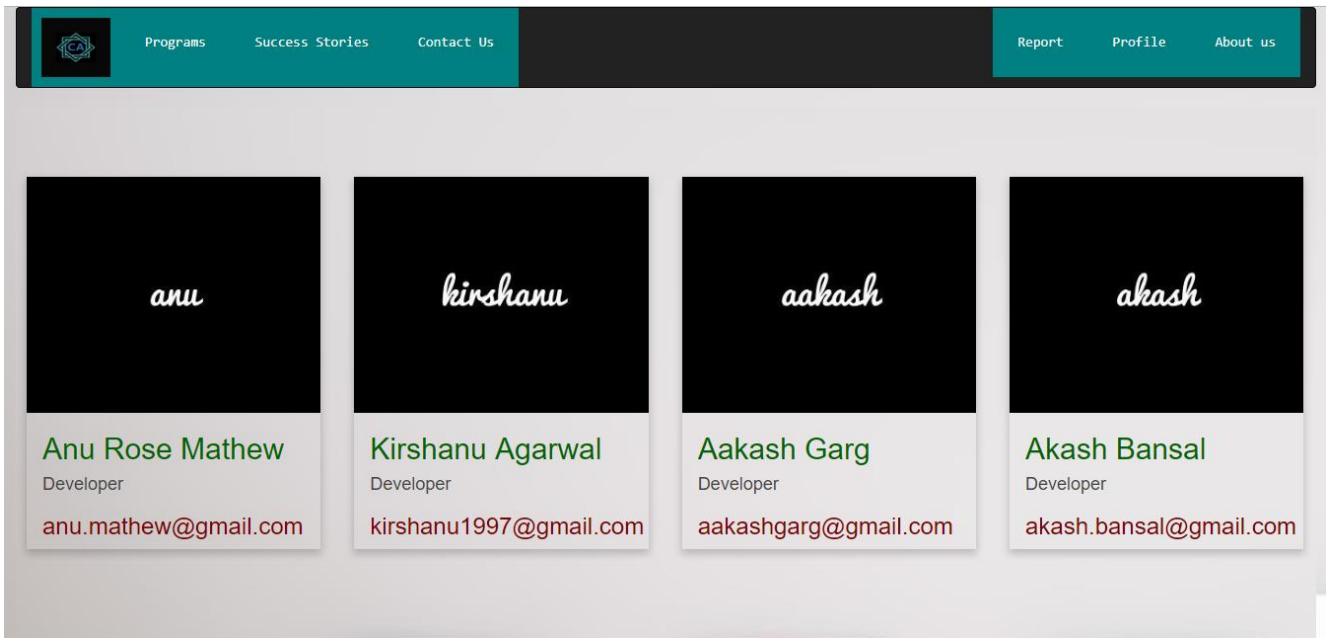
8.9 Profile view page



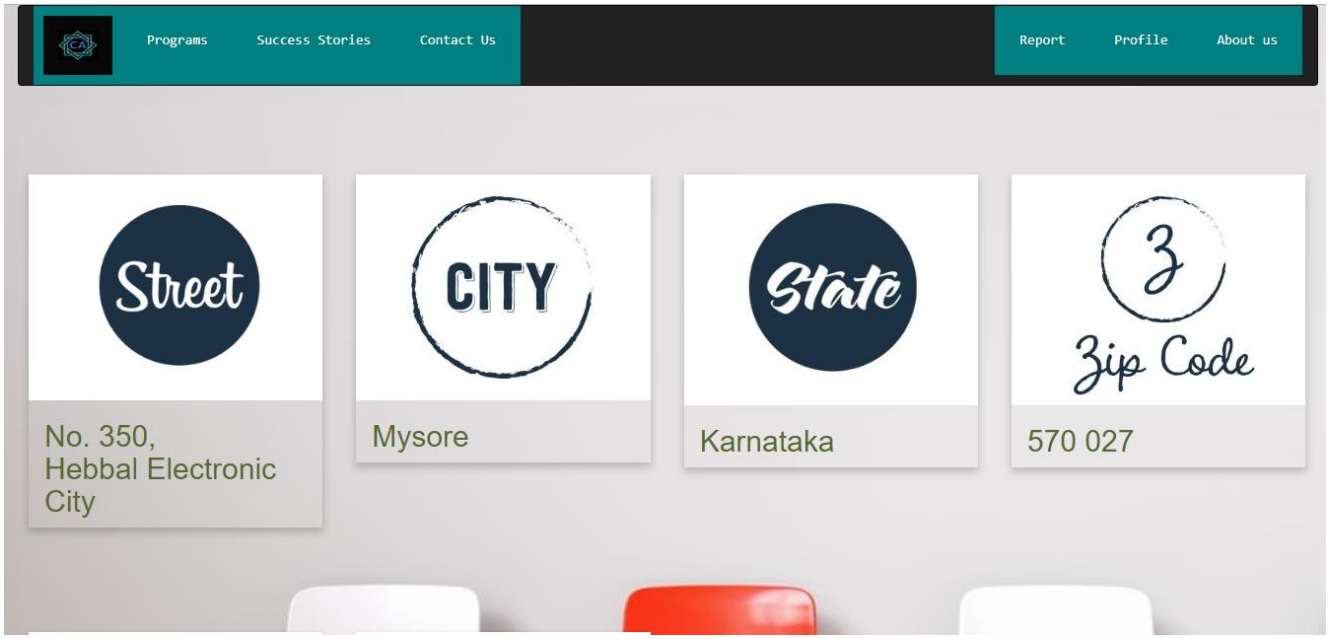
8.10 after clicking logout



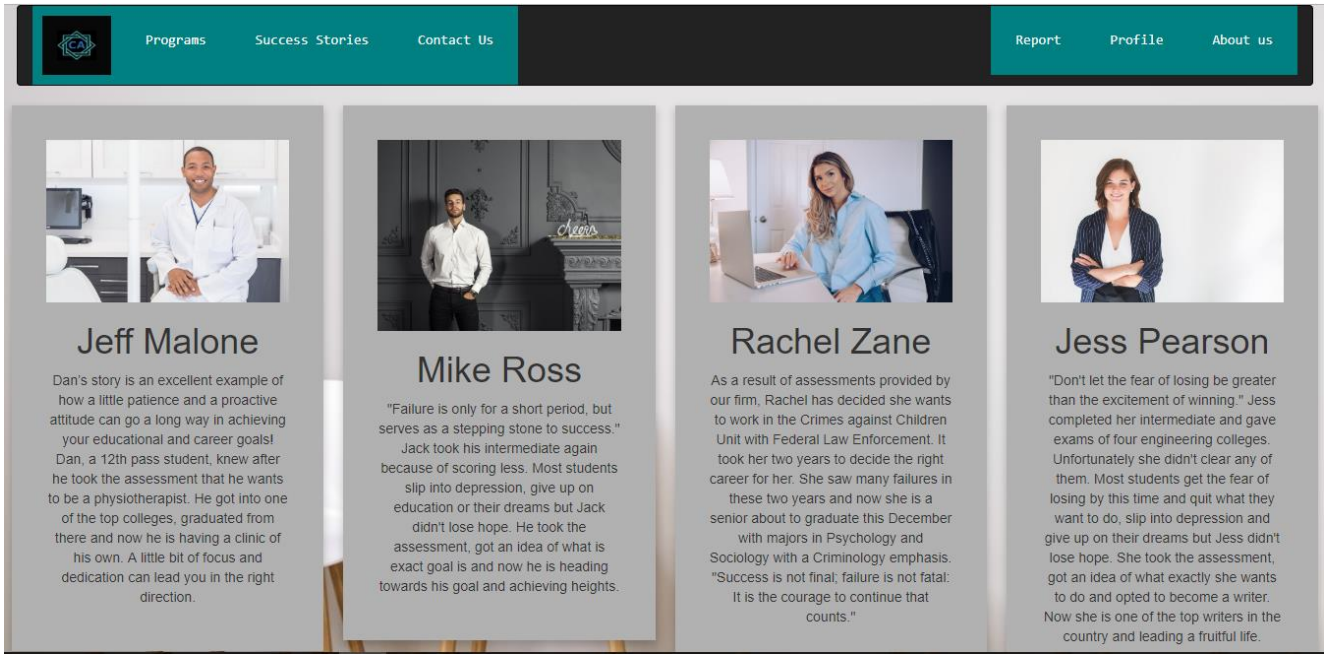
8.11 about company



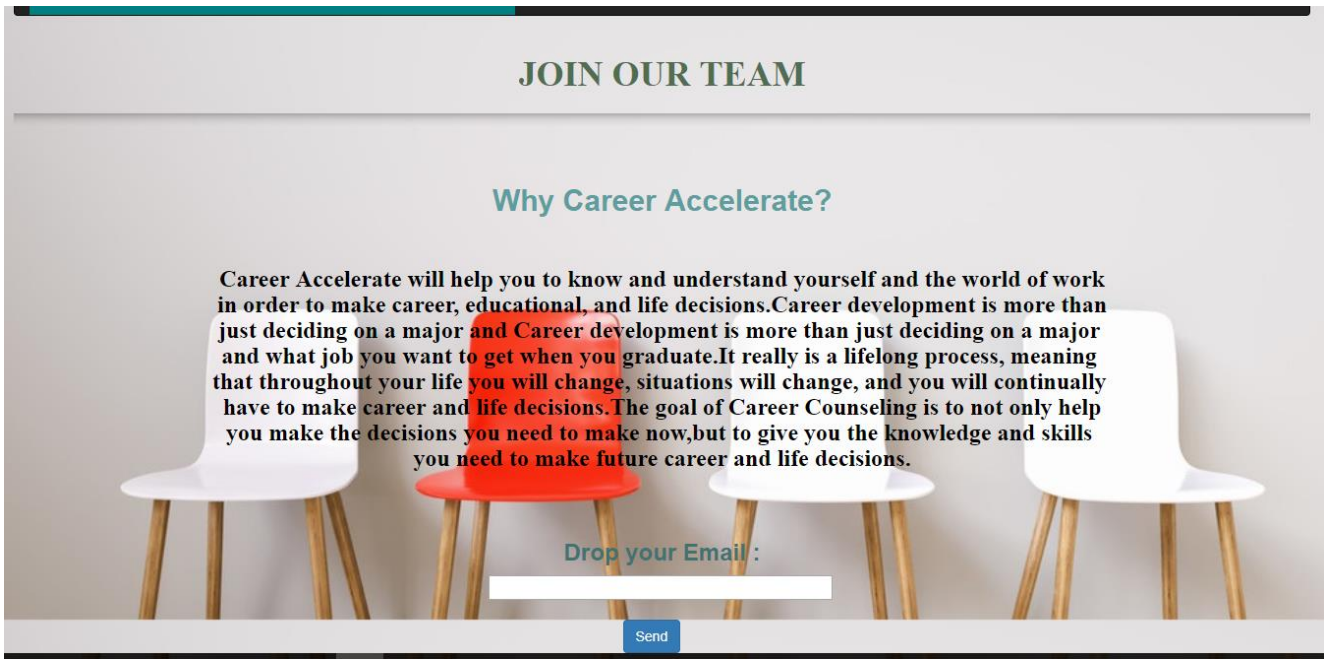
8.12 about team



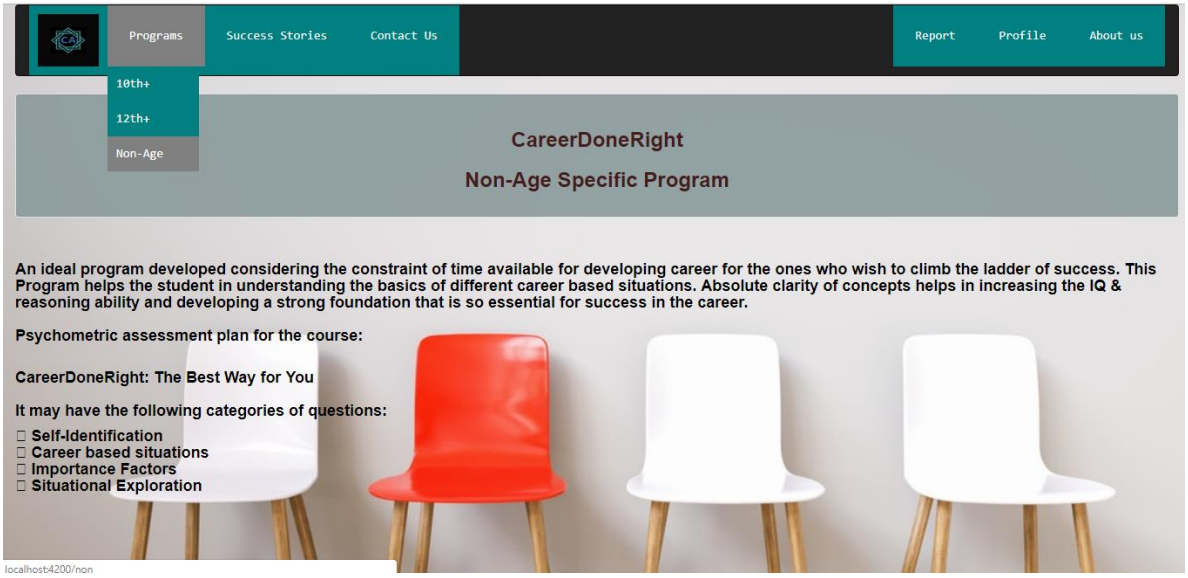
8.13 contact us



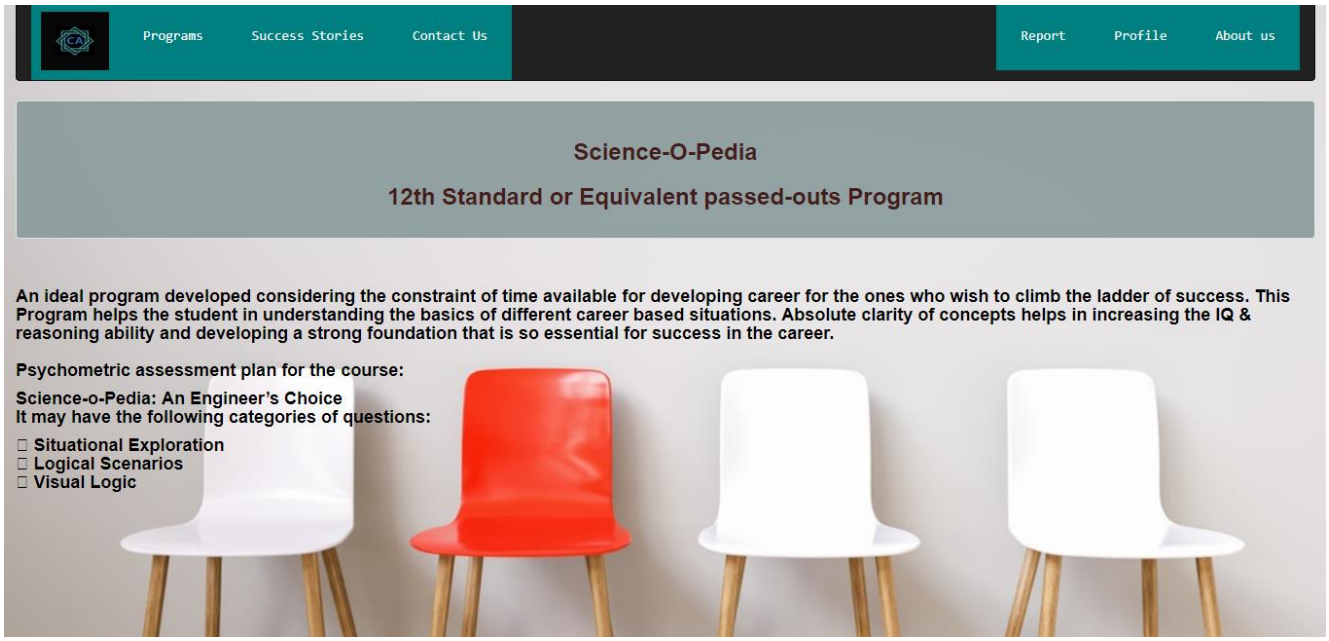
8.14 success stories



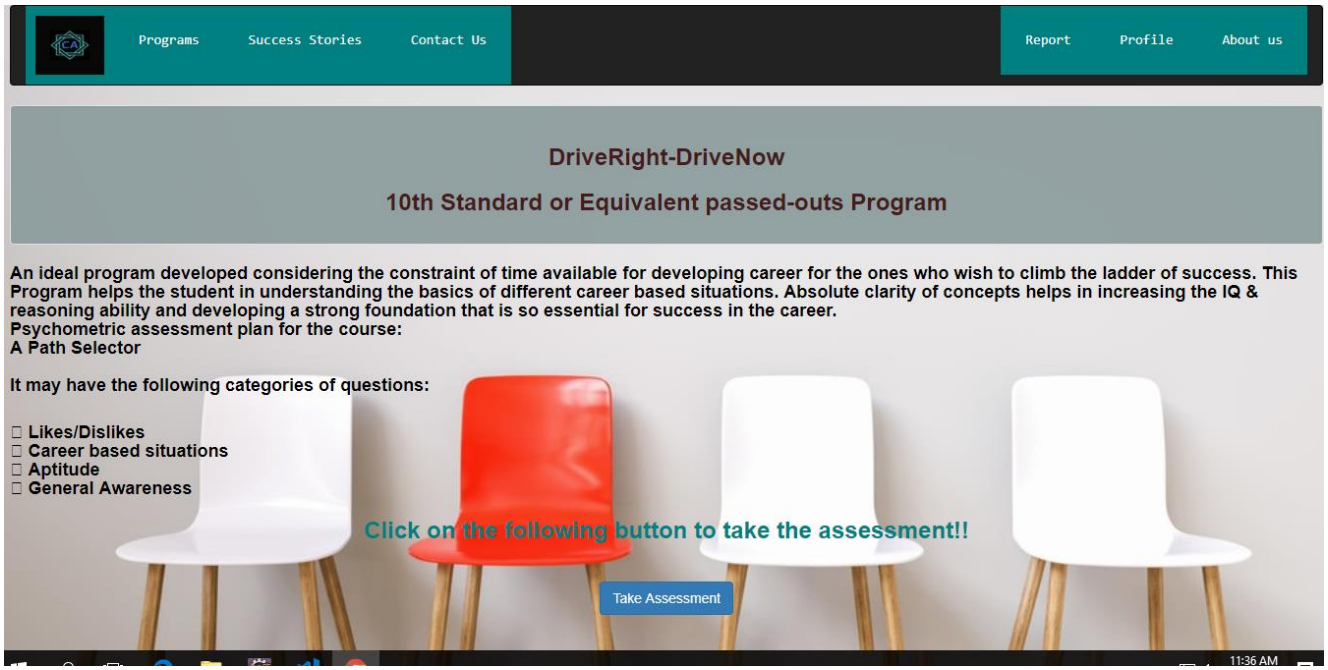
8.15 join our team



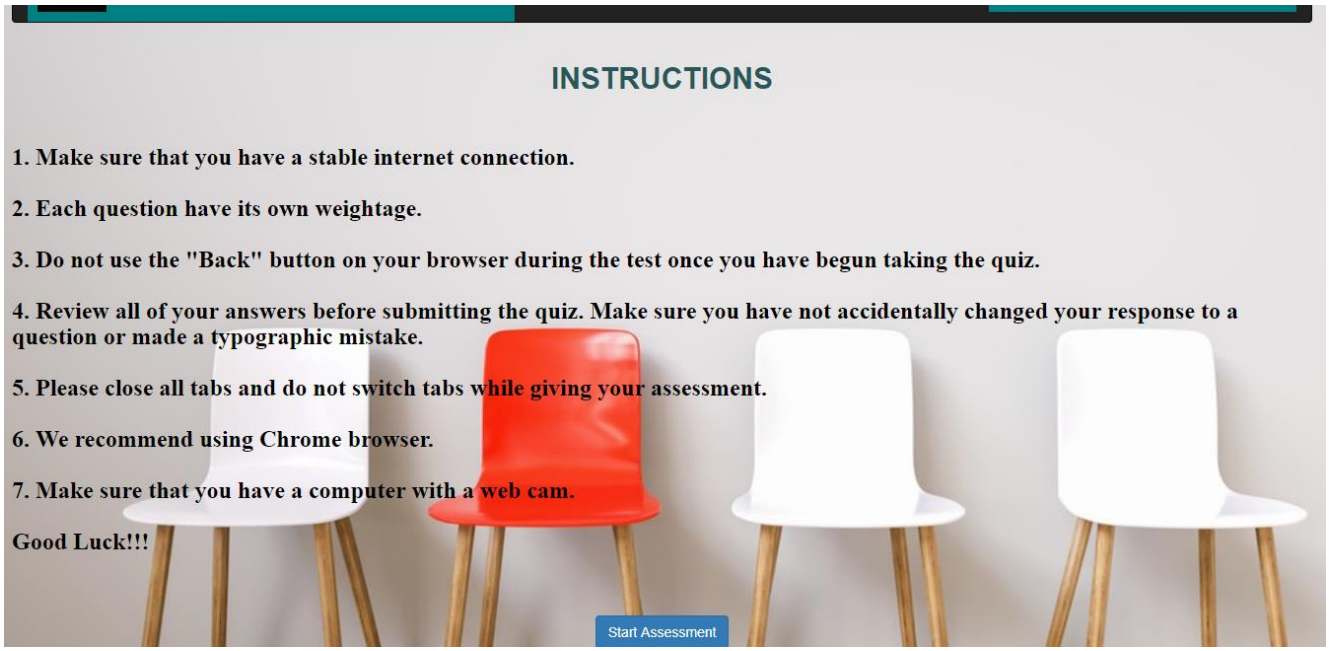
8.16 non age assesment



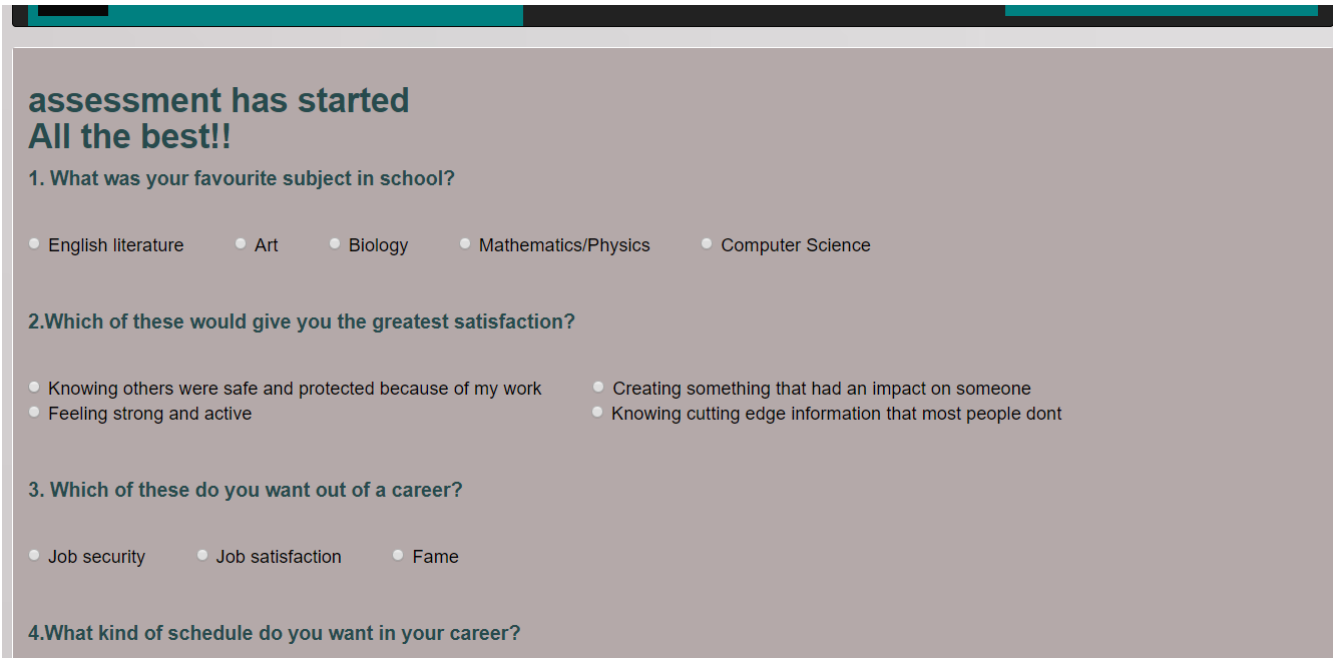
8.17 12th assessment



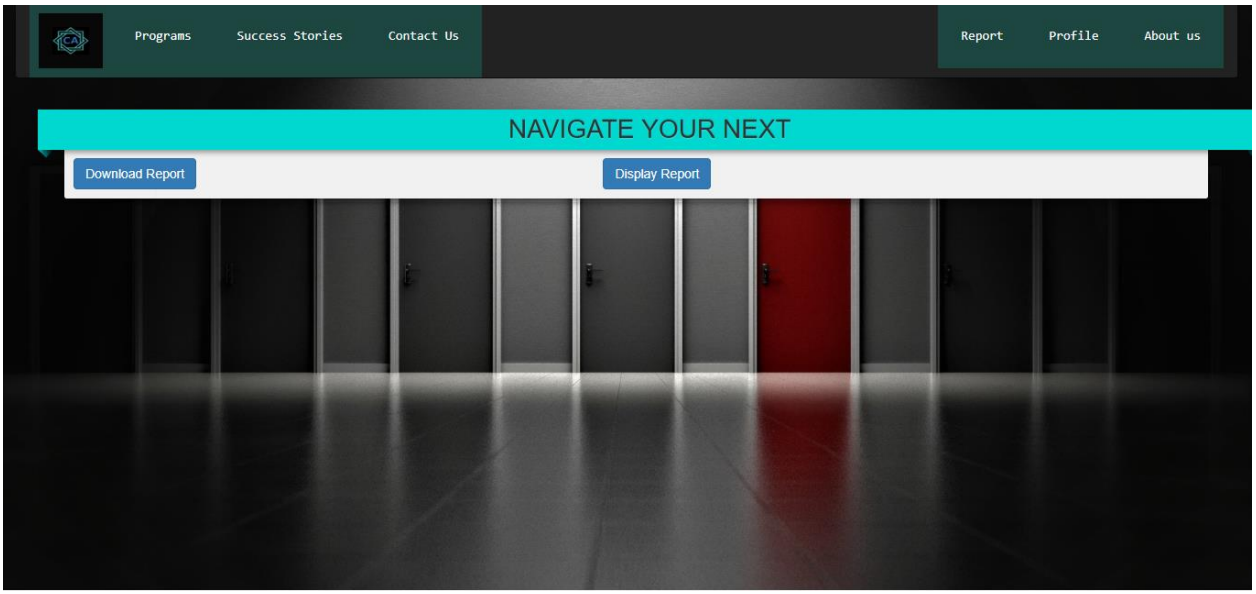
8.18 10th assessment



8.19 Instruction page

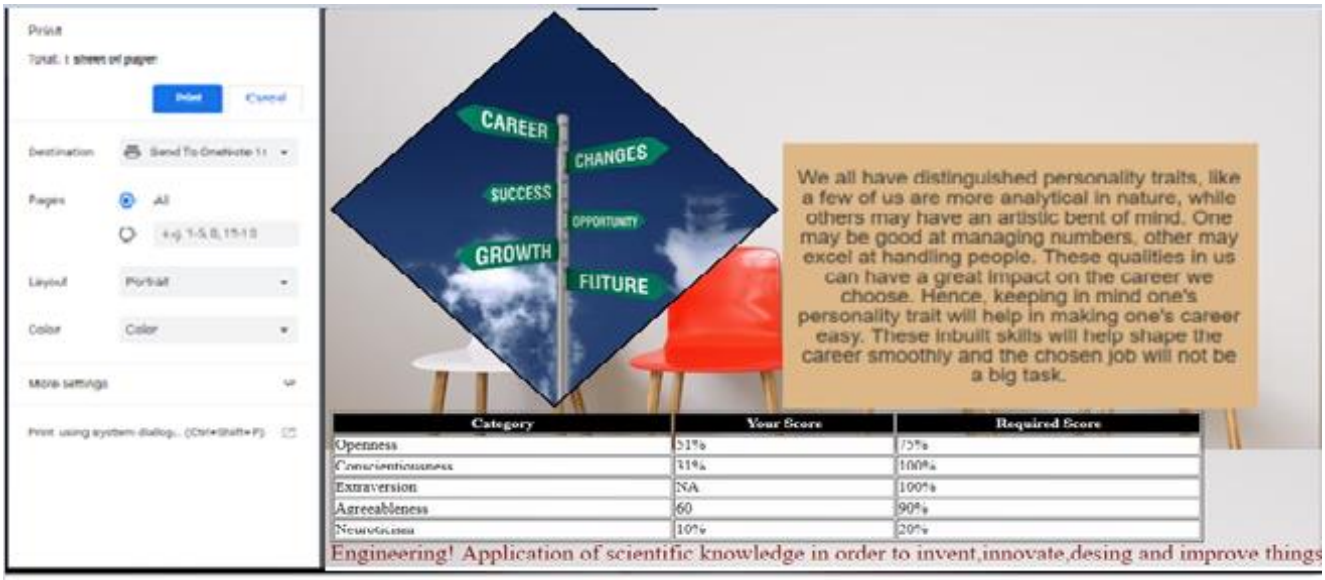


8.20 taking assessment



8.21 reportsceen

8.22 displaying report



8.23 downloading report