

ACKNOWLEDGEMENT

We express our unfathomable sense of pleasure towards **Dr Prof GargiRajpara** Principle of **LDRP-ITR**, who gave us a chance to work. She frequently came to motivate and provided such an extraordinary infrastructure and resources to work in her premises.

We express our deep sense of gratitude and indebtedness to our internal guide **Prof Riya Gohel** for accepting us to work under her training and supervision. He took prolonged interest in our work and directed to us toward the predefined goal. He has shown us a way to pursue excellence. He has been a big factor of motivation in our project.

We are also very thankful to our HOD of CE/IT department **Dr. Prof. HirenB Patel** and all the faculty members who provided to be a constant motivation for the knowledge acquisition and the moral support throughout the whole project.

We sincerely thank our Computer Engineering Department, **LDRP-ITR** for the academic advancement it has provide to us during the whole engineering course and provided an opportunity for project training.

Last but not the least my heartily thanks is to my friends and batch mates, who have provided us with innumerable discussions on many technicalities and friendly tips.

We would like to express our heartiest gratitude to our family and friends who has always guided us towards the path of success and those entire people who have directly or indirectly helped us in making this project work successful.

Bhatt Aayush (1620BECE30065)

Aashay Gandhi (1620BECE30064)

Akshay Chauhan (1620BECE30073)

Jaimin Raval (1620BECE30110)

Abstract

Our project provides Developers a option to make Web pages for performing CRUD operations by providing code to them from the input which we take as field name and field type name and can run on their own system.t uses Alexa to get input from the user such as project name, modules and forms, then generate the code as per the input and store the code in AmazonS3 then the code generated is mailed to the user on their registered email. Then the user can run the code on his system in Eclipse IDE by using his database.

TABLE OF CONTENTS:

- Acknowledgement
- Abstract
- Table of Contents9
- List of Figures
- List of Tables
- 1 Introduction**
 - 1.1 Introduction
 - 1.2 Scope
 - 1.3 Project summary and Purpose
 - 1.4 Overview of the project
 - 1.5 Problem definition
- 2 Technology and Literature Review**
 - 2.1 About Tools and Technology
- 3 System Requirements Study**
 - 3.1 User Characteristics
 - 3.2 Hardware and Software Requirements
 - 3.3 Constraints
 - 3.3.1 Regulatory Policies
 - 3.3.2 Hardware Limitations
 - 3.3.3 Parallel Operations
 - 3.3.4 Technical Constraints
 - 3.3.5 Reliability Requirements
 - 3.3.6 Safety and Security Consideration
 - 3.4 Assumptions and Dependencies
- 4 System Analysis**
 - 4.1 Study of Current System
 - 4.2 Problem and Weaknesses of Current System
 - 4.3 Requirements of New System
 - 4.3.1 User Requirements
 - 4.3.2 System Requirements
 - 4.4 Feasibility Study
 - 4.4.1 Does the system contribute to the overall objectives of the organization?
 - 4.4.2 Can the system be implemented using the current technology and within the given cost and schedule constraints?
 - 4.4.3 Can the system be integrated with other systems which are already in place?
 - 4.5 Requirements Validation(is concerned with showing that the requirements actually define the system which the customer wants)

4.6 Activity/Process In New System(Use event table)

4.7 Class Diagram

4.8 Activity Diagram

4.9 Use case Diagram

4.10 Sequence Diagram

4.11 Data Flow Diagram

5 System Design

5.1 System Application Design

5.1.1 Method Pseudo code

5.2 Database Design/Data Structure Design

5.2.1 Table and Relationship

5.3 Input/Output and Interface Design

5.3.1 Samples Of Forms, Reports and Interface

6 System Testing

6.1 Purpose

6.2 Scope of testing

6.3 Test plan

6.4 Test cases

7 Conclusion and Future Possibility

Bibliography

List of Figures:

Fig 1: Iterative Process Model

Fig 2: Class Diagram

Fig 3: Project Schedule

Fig 4: Class Diagram

Fig 5: Activity Diagram

Fig 6 Sequence Diagram

Fig 7:DFD level 0

Fig 8: DFD level 1

Fig 9: DFD level 2

Fig i: Home Screen

Fig ii: Emergency Number

Fig iii: Contact list

Fig iv: Service History

List of tables:

Fig a) Contacts list

Fig b) Service history

Fig c) Emergency Contacts

Fig d) Test Case Table

1) Introduction:

1.1 Introduction:-

The project is to develop an web application that will help in accessing the data from the developer main Smartphone or computer without being fear of losing phone anywhere, This all data accessing will be handled by the interface of our web application..

1.2 Scope:-

Our project provides Developers a option to make Web pages for performing CRUD operations by providing code to them from the input which we take as field name and field type name and can run on their own system.

1.3 Project Summary and purpose:

It uses Alexa to get input from the user such as project name, modules and forms, then generate the code as per the input and store the code in AmazonS3 then the code generated is mailed to the user on their registered email. Then the user can run the code on his system in Eclipse IDE by using his database.

1.4 Overview of project:

We have chosen the iterative Model for this system development. An iterative lifecycle model does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which can then be reviewed in order to identify further requirements.

This process is then repeated, producing a new version of the software for each cycle of the model.

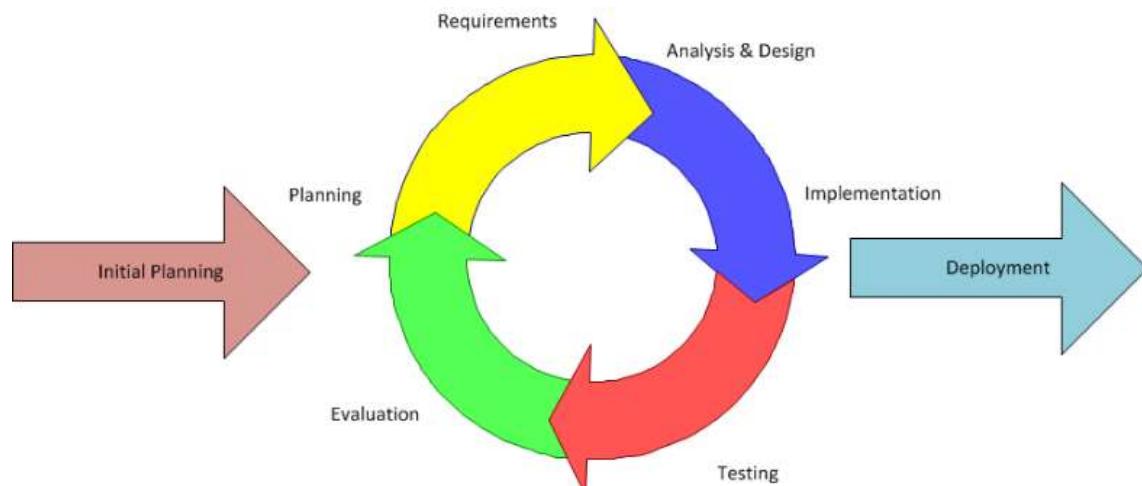


Fig 1 Iterative Process Model

Consider an iterative lifecycle model which consists of repeating the following four phases in sequence:

Requirements phase, in which the requirements for the software are gathered and analyzed. Iteration should eventually result in a requirements phase that produces a complete and final specification of requirements.

Design phase, in which a software solution to meet the requirements is designed. This may be a new design, or an extension of an earlier design.

Implementation and Test phase, when the software is coded, integrated and tested.

Evaluations phase in which the software is evaluated, the current requirements are reviewed, and changes and additions to requirements proposed.

For each cycle of the model, a decision has to be made as to whether the software produced by the cycle will be discarded, or kept as a starting point for the next cycle (sometimes referred to as incremental prototyping). Eventually a point will be reached where the requirements are complete and the software can be delivered, or it becomes impossible to enhance the software as required, and a fresh start has to be made.

The iterative lifecycle model can be likened to producing software by successive approximation. Drawing an analogy with mathematical methods that uses successive approximation to arrive at a final solution, the benefit of such methods depends on how rapidly they converge on a solution.

The key to successful use of an iterative software development lifecycle is rigorous validation of requirements, and verification (including testing) of each version of the software against those requirements within each cycle of the model.

The first three phases of the example iterative model is in fact an abbreviated form of a sequential V or waterfall lifecycle model. Each cycle of the model produces software that requires testing at the unit level, for software integration, for system integration and for acceptance.

As the software evolves through successive cycles, tests have to be repeated and extended to verify each version of the software.

1.5 Problem Definition:

Source Code Generator

We are living in a world where everything is happening at the speed of a Formula-1 car in a race track. Keeping tick of essentials in this fast-paced life sometimes turns out to be quite demanding.

Whenever a developer wants to build a website , apart from patterns, ideas , the main headache is to do coding for certain webpages or we can say standard web pages like registration forms , login forms , cookies message and other stuff which the user encounter in all types of websites.

Source Code Generator is a web application in which the developer just has to add description of the webpage.

All types of codes i.e frontend and backend of that particular web page will be generated by our application as per the description of the developer

Some minor changes can also be done by the developer using Wordpress after the webpage is generated along with its code by our application

Since the developer has his code now , code can also be edited by the developer for the changes he needs in his webpage.

2. Technology and Literature Review:

2.1 About Tools and Technology:

Lambda: AWS Lambda is an [event-driven](#), serverless computing platform provided by [Amazon](#) as a part of the [Amazon Web Services](#). It is a computing service that runs code in response to [events](#) and automatically manages the computing resources required by that code. It was introduced in November 2014.^[1]

The purpose of Lambda, as compared to [AWS EC2](#), is to simplify building smaller, on-demand applications that are responsive to events and new information. AWS targets starting a Lambda instance within milliseconds of an event. [Node.js](#), [Python](#), [Java](#), [Go^{\[2\]}](#), [Ruby^{\[3\]}](#) and [C#](#) through [.NET Core](#) are all officially supported as of 2018, and other languages can be supported via call-outs.

AWS Lambda supports securely running [native Linux executables](#) via calling out from a supported runtime such as Node.js.^[4] For example, [Haskell](#) code can be run on Lambda.^[5]

AWS Lambda was designed for use cases such as image or object [uploads](#) to Amazon S3, updates to [DynamoDB](#) tables, responding to website clicks or reacting to sensor readings from an [IoT](#) connected device. AWS Lambda can also be used to automatically provision back-end services triggered by custom [HTTP requests](#), and "spin down" such services when not in use, to save resources. These custom HTTP requests are configured in AWS API Gateway, which can also handle [authentication](#) and [authorization](#) in conjunction with [AWS Cognito](#).

Unlike Amazon EC2, which is priced by the hour but metered by the second, AWS Lambda is metered in increments of 100 milliseconds. Usage amounts below a documented threshold fall within the AWS Lambda free tier - which does not expire 12 months after account signup, unlike the free tier for some AWS services.^[6]

Other similar solutions in the market are [Google Cloud functions](#), [Oracle Cloud Fn](#) and [Azure Functions](#).

3 System Requirements Study

3.1 User Characteristics:

The system has 2 actor that the one is the Alexa who will provide information to the user as Alexa is user interface between the system side and client or user side. Alexa provide information in two ways as user can speak and data is entered in to the form or user writes the data according to the form.

3.2 Hardware and Software Requirements

For development:

Criterion	Description
OS version	Windows XP, Windows 7, Windows 8 and higher version
	GNOME, KDE or Unity desktop on Ubuntu or Fedora or GNU/Linux Debian
RAM	3 GB RAM minimum, 4 GB RAM recommended
Disk space	500 MB to 1GB disk space
Space for Android SDK	At least 2 GB for Android SDK, emulator system images, and caches
JDK version	Java Development Kit (JDK) 7 or higher
Screen resolution	1280×800 minimum screen resolution

For app

API level	20 or above
RAM	2 GB RAM minimum, 3 GB RAM recommended
API Size	2.5 MB
Space required	40 MB
Permissions	<ul style="list-style-type: none">● Contacts● Your Location● SMS● Storage.● Email
Screen resolution	1280×800 minimum screen resolution

3.3 Constraint:

3.3.1 Regulatory Policies:

The code written on Amazon web services is basically runs on Lambda which is serverless computation compiling. Aws Lambda executes your code only when needed and scaled automatically, with Aws lambda, you can run code for virtually any type of application or backend service-all with zero administration.

If you choose to use my Service, then you agree to the collection and use of information in relation to this policy. The Personal Information that I collect is used for providing and improving the Service. I will not use or share your information with anyone except as described in this Privacy Policy.

If you choose to use my Service, then you agree to the collection and use of information in relation to this policy. The Personal Information that I collect is used for providing and improving the Service. I will not use or share your information with anyone except as described in this Privacy Policy.

The terms used in this Privacy Policy have the same meanings as in our Terms and Conditions, which is accessible at mail on your registered email-id unless otherwise defined in this Privacy Policy.

Java-Eclipse:Eclipse is an IDE(Integrated Development Environment)is a computer programming,and in 2014 it one of the most widely used in java website poll.It contains a base workspace and an extensible plug-in system for customizing environment.Eclipse is mostly used in Java and its primary use to develop Java Applications,but it may also used to develop applications in other programming languages

Service Provider:-

- To provide the Service on our behalf;
- It allows you to download the form which is created using Crud operations
- It provide generated web pages with Back-End
- To provide Feedback
- To solve queries asked during form registration
- Using Remote Database Service user can access the form at any place
- It provides cloud Based services

Security

I value your trust in providing us your Personal Information, thus we are striving to use commercially acceptable means of protecting it. But remember that no method of transmission over the internet, or method of electronic storage is 100% secure and reliable, and I cannot guarantee its absolute security.

Links to Other Sites

This Service contains no links to other sites.

Children's Privacy

These Services do not address anyone under the age of 13. I do not knowingly collect personally identifiable information from children under 13 If you are a parent or guardian and you are aware that your child has provided us with personal information, please do the necessary .

Changes to This Privacy Policy

I may update our Privacy Policy from time to time. Thus, you are advised to review this page periodically for any changes. I will notify you of any changes by posting the new Privacy Policy on this page. These changes are effective immediately after they are posted on this page.

Contact Us

If you have any questions or suggestions about my Privacy Policy, do not hesitate to contact me via email-id

This privacy policy page was created at privacypolicytemplate.net and modified/generated by [App Privacy Policy Generator](#)

3.3.2 Hardware Limitations

The Screen size must be 1280×800 for the best functionality. Ram must be 3GB for best functionality. The phone must be GPS Enabled.

3.3.3 Parallel Operations

Only one operation can be performed at a time as all the data are to be conveyed through the Email Internally if more than one operations can be performed, as for accessing the details if there exists more than one contact of same name then all the details entered by that user will be accessed.

3.3.4 Technical Constraints:

- Operating System/Platform Supported: The Application will work only on computer and not on IOS,QT or Solaris.
- The Email-id entered during form registration and filled during form registration should be matched
- All the permissions must be granted for best outcome.
- Language Constraint: Available only in English and not in any other language.

3.3.5 Reliability Requirements:

System should be reliable. It should keep secure all the information regarding to any user. The system must preserve the stored data

3.3.6 Safety and Security Considerations:

The user is responsible for the safety his/her data in case he/she is both the admin and the user. Otherwise the admin is responsible as all the data is accessed through OTP and can only be sent to the list approved by the admin.

3.3.6 SECURITY:

The Information should be Secure, there should not be any kind of malfunctioning. All the data, staff profiles must be stored securely in the system. System Information will not be changed by any person who is unauthorized.

3.3.7 FLEXIBILITY:-

System should be quite flexible to install and maintain.

3.3.8 EFFICIENCY:-

System should be efficient enough to meet all kinds of requirements as required by all the users. The system should not hang or lose its efficiency in any Kind of worse conditions. It should provide the correct output in all manners.

3.3.9 USER FRIENDLINESS:-

System should be user friendly, so that any user can access the system.

3.4 Assumptions and Dependencies

Assumptions:

- The Admin adding the person in accessing list will be a trusted user.
- All the required permissions and hardware, software requirements will be granted.
- Following the given commands to access the data of the Alexa.
- In background the process will be running and no other process should stop it.
- One Time Password service will be available to the user specified email id.
- Only specified number from the accessing list will access the data.

Dependencies:

- All the function is depended on the SIM for functionality.
- The Phone must be GPS Enabled.

4 System Analysis:

4.1 Study of Current System:

Recently the Smartphone are becoming totally dependent on the internet. Hence if there is no internet then they are almost of no use, currently there are very few systems that can access data from the mobile WITHOUT internet, if they want contact or anything else then if there is no internet then it a problem to get the required data but it can be done through the SMS facility if correct application is created for it. Hence our view is to create an Application for this that can serve 3 functions contacts accessing, mobile last location and mobile locating through alarm.

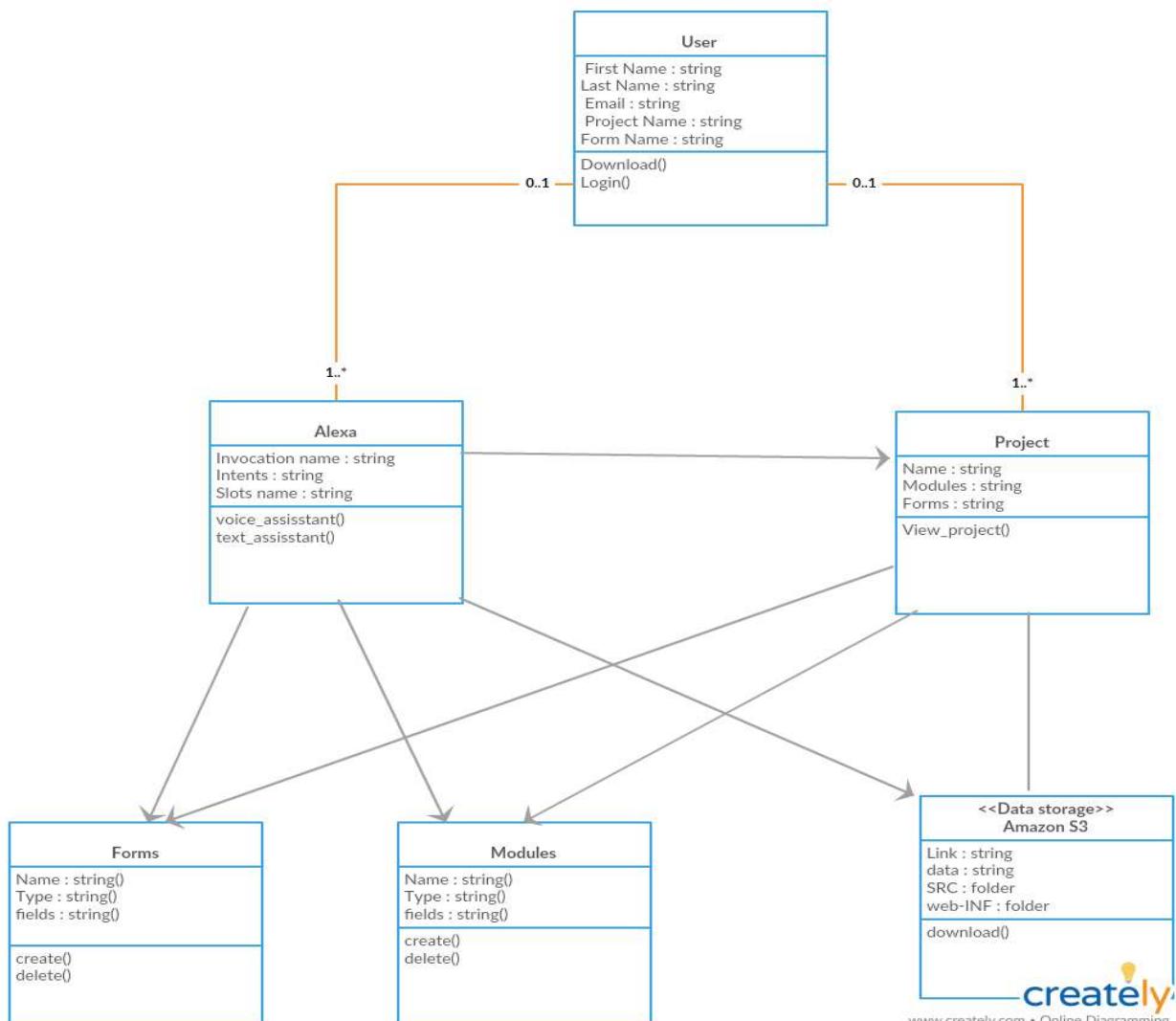


Fig 2 Class Diagram

4.2 Problem and weakness of current system

The only problem and weakness of the current system will be if the developer wants to make dynamic web pages he won't be able to make it through our System. You cannot make updates to your webpages. You can't change the backend processes.

4.3 Requirements of New System

4.3.1 User Requirement:

This is concerned with specifying and software that will successfully satisfy the user requirement the technical needs of the system may vary considerably, but might include:

- The facility to produce outputs in given time.
- Response time under certain conditions.
- Ability to process a certain volume of tasks at a particular speed.
- Facility to communicate data to distant location

4.3.2 System Requirement:

API level	20 or above
RAM	2 GB RAM minimum, 3 GB RAM recommended
Web API Size	13 MB
Space required	40 MB
Permissions	<ul style="list-style-type: none">• Contacts• Your Location

	<ul style="list-style-type: none"> ● SMS ● Storage.
Screen resolution	1280×800 minimum screen resolution

4.4 Feasibility Study

It is considered as an important stage of a project. It can be described as an assessment of the practicality of a proposed project. The main goal of feasibility study is to test the technical, social and economic feasibility of the system to be developed. A feasibility study is an analysis of the viability of an idea. A well-designed feasibility study should provide a historical background of the business or project, a description of the product or service.

Mainly three key considerations are involved in the feasibility analysis.

Economical Feasibility

Technical Feasibility

Social Feasibility

4.4.1 Social Feasibility: Does the system contribute to the overall objectives of the organization?

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user.

There are 3 main objectives of our organization and this contributes to all of the 3 objectives.

4.4.2 Economical Feasibility: Can the system be implemented using the current technology and within the given cost and schedule constraints?

This also called as Cost and Benefit Analysis. Cost and benefit analysis gives us for computerization, the cost will reduce in some aspects and respective increase in not as much as compared to reduction computerization will reduce the requirement staff. It will also reduce the cost of non-reusable stationary; the computer stationary will replace the various types of bills and registers.

Yes, As we are determined to create an android application for these objectives and today mostly all the current technologies that are to be created in android can be implemented today by Android Studio and no other technology is required for this implementation.

4.4.3 Technical Feasibility: Can the system be integrated with other systems which are already in place?

In this type of feasibility we have to see that whatever existing system in the organization supports the computerized systems or not. In other words, is the computerized system working same as that of the existing system? The computerized system may not create any problem, if any problem regarding to the system occurs, then the manner can contact to the software consultancy firm, so that they can remove the problems or bugs

The system is created in modules hence can be integrated with other technologies as well as each module as its own behavior but for any other person to get the module they have to get in contact with our developer.

4.4.4 TIME FEASIBILITY

Time feasibility of the study is if the project can be implemented in a frame of time or not, in our study our project can be implemented in approx 75 days.

4.5 REQUIREMENTS VALIDATION

Requirement Validation examines the specification to ensure that all the system requirements have been stated unambiguously, those inconsistencies, errors have been detected and corrected and the work products confirm to the standard.

Source of the requirements are identified. Final statement of requirement has been examined by original source.

Requirements related to main requirements are found.

Requirements are clearly stated and are not misinterpreted.

All sources of requirements are covered to get maximum requirement.

All methods of finding requirements are applied.

Requirements are not duplicated and each of them gives distinct idea of processes within project.

Requirement associated with system performance, behavioral and operational characteristics are clearly stated.

Each requirement is being analyzed to prove its feasibility for the current system

4.6 ACTIVITY/PROCESS IN NEW SYSTEM (Use event table)

We can say that scheduling is a very critical part of any project to manage it smoothly. A schedule is a listing of a project's milestones, activities, and deliverables, usually with planned start and finish dates. This helps in achieving the goal easily and faster. We can generate schedule with the help of many charts or graphs.

Task Name	Duration	Start	Finish
Searching and analysis	15 Days	14-05-2019	29-05-2019
Feasibility Study	10 Days	30-05-2019	09-06-2019
System Analysis	30 days	09-06-2019	09-07-2019
Design	12 days	09-07-2019	21-07-2019
Implementation	5 days	22-07-2019	27-07-2019
Testing	5 days	27-07-2019	02-08-2019



Fig 3: Project Schedule

4.7 CLASS DIAGRAM:

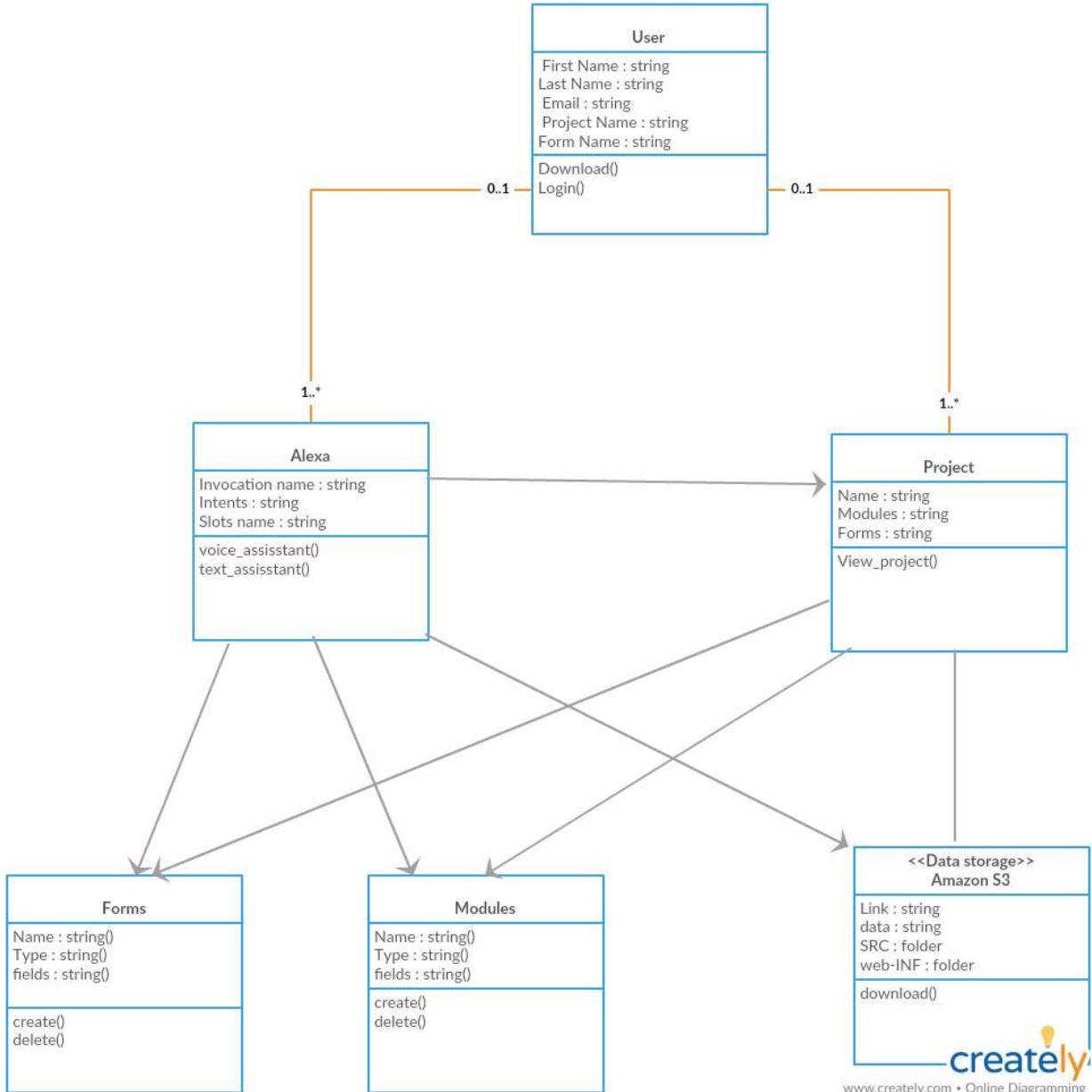


Fig 4: Class Diagram

4.8 Activity Diagram:

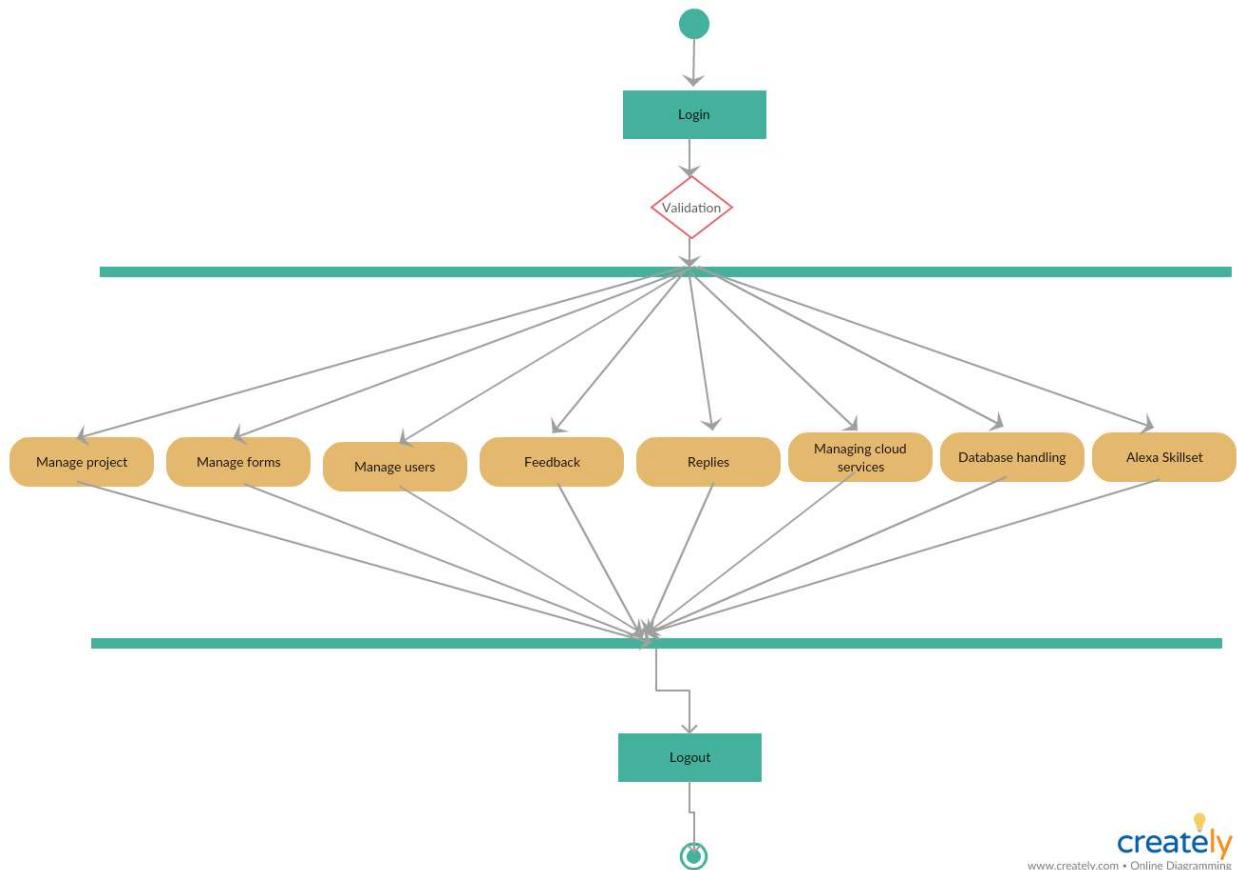
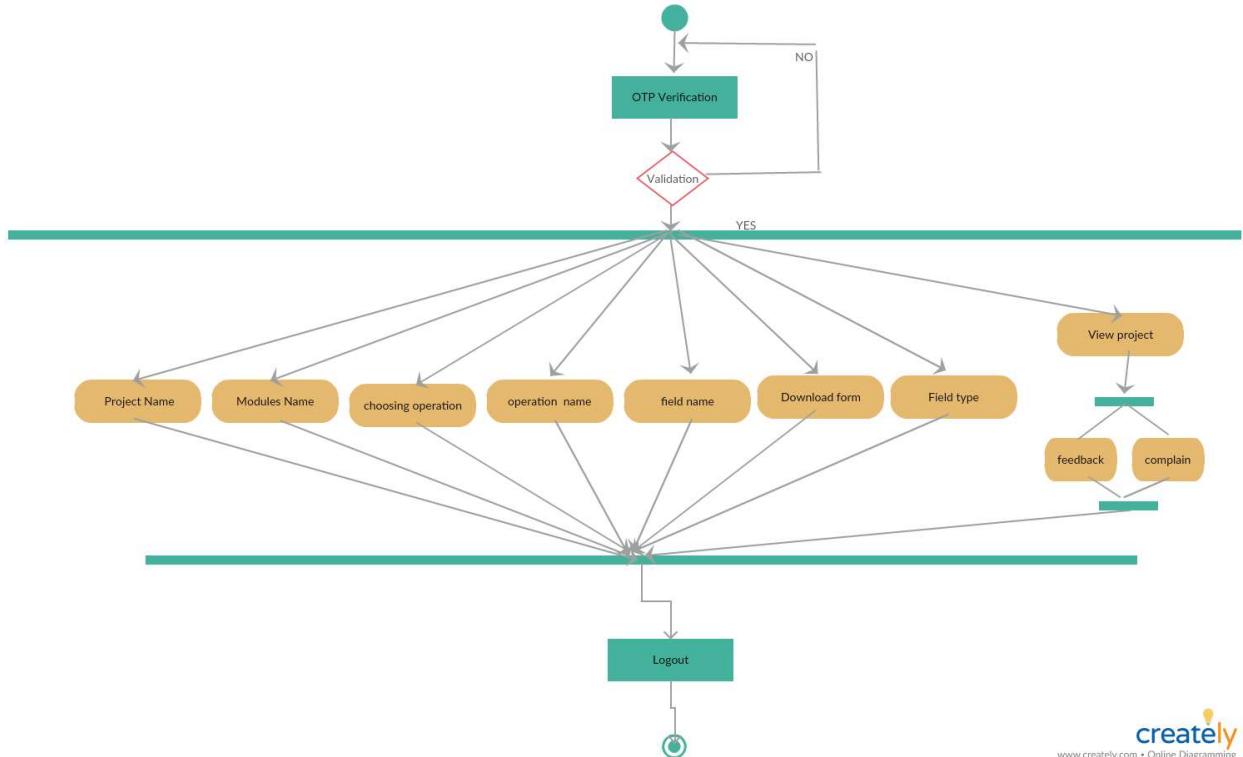


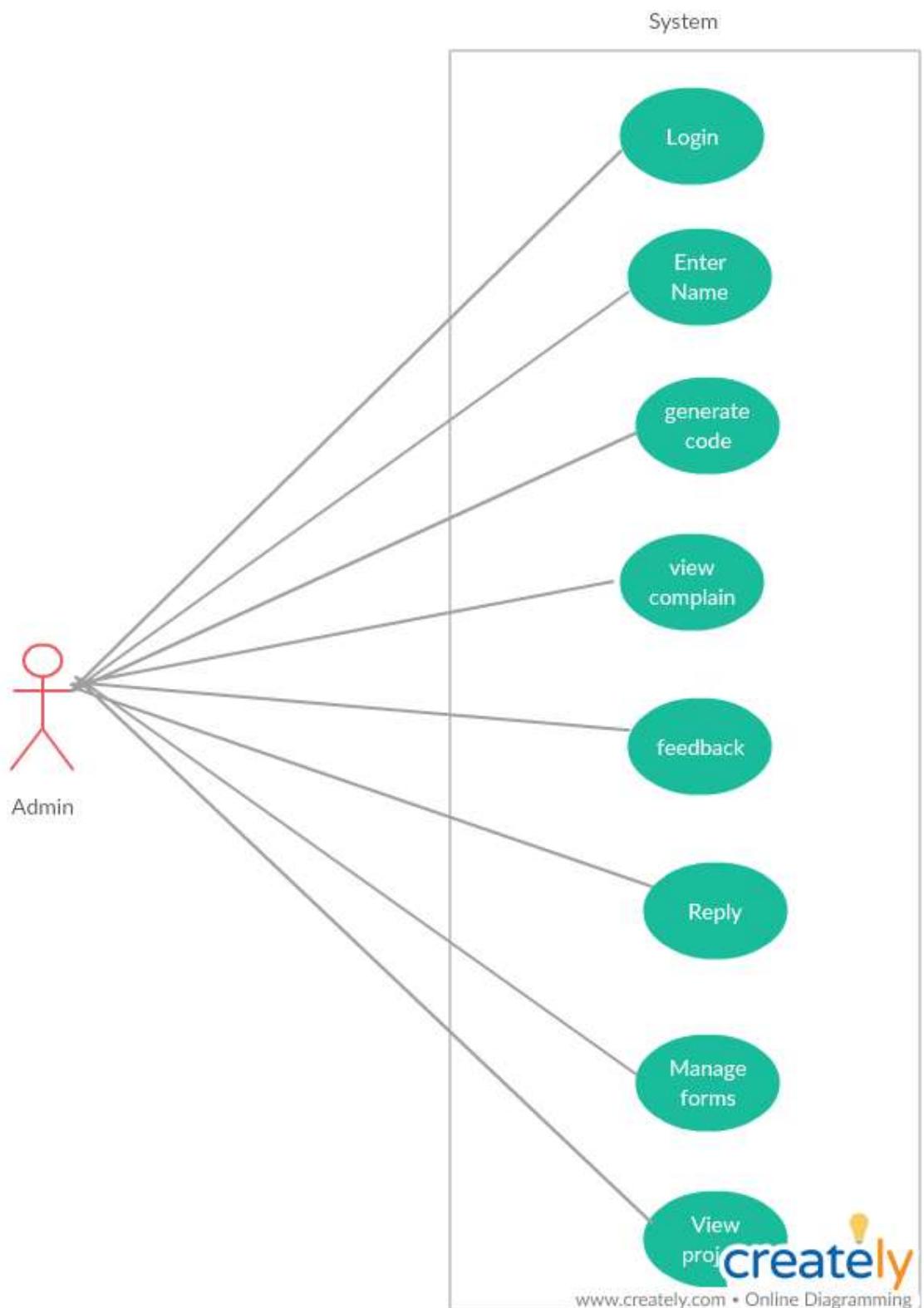
Fig 5: Activity Diagram

Admin Activity Diagram

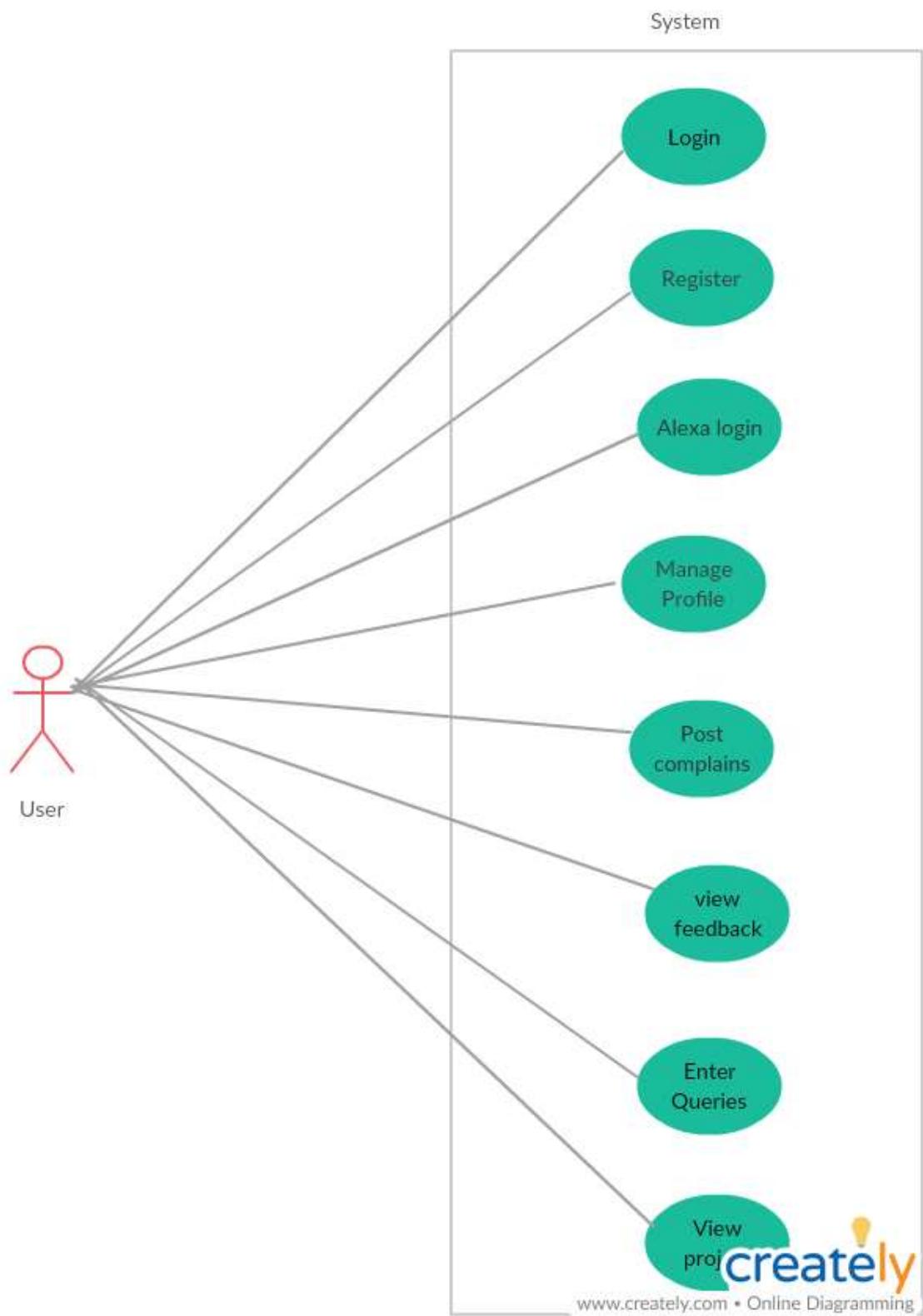


User Activity diagram

4.9 Use case Diagram:



Usecase Diagram For Admin



Usecase DiagramFor User

4.9 Sequence Diagram:

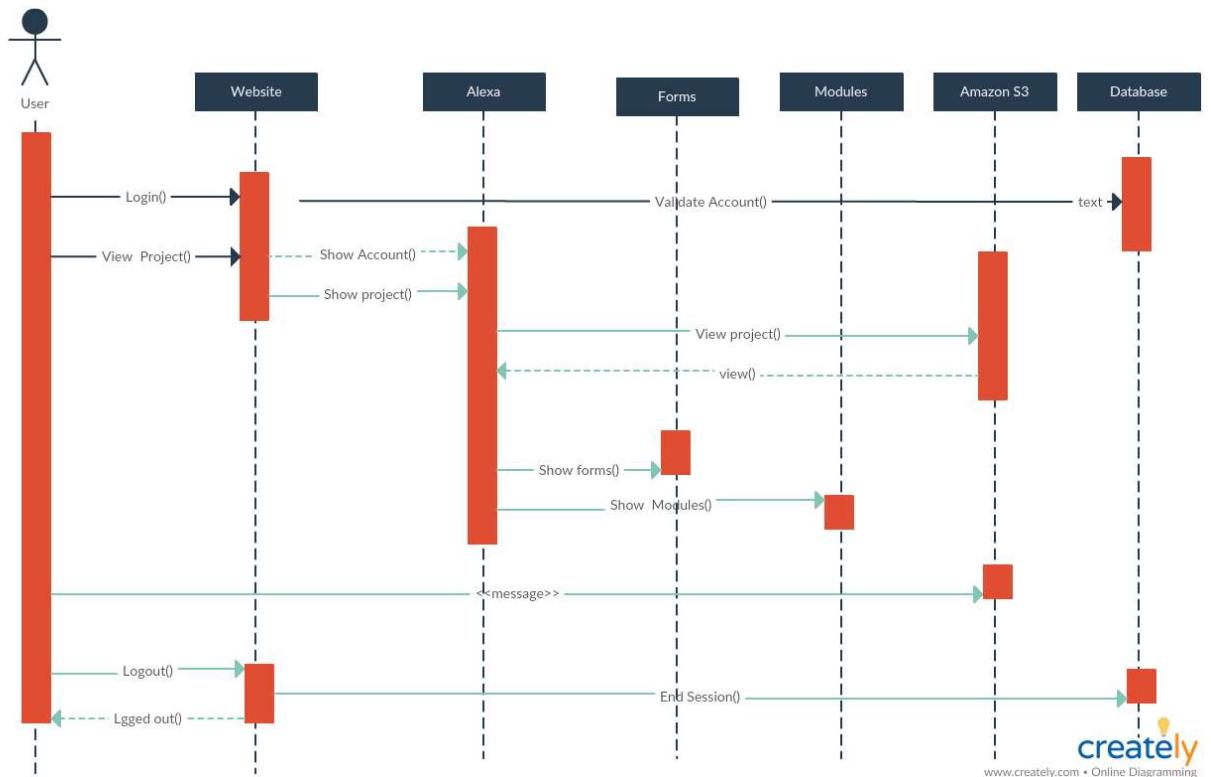


Fig 6: Sequence Diagram

4.10 Data Flow Diagram:

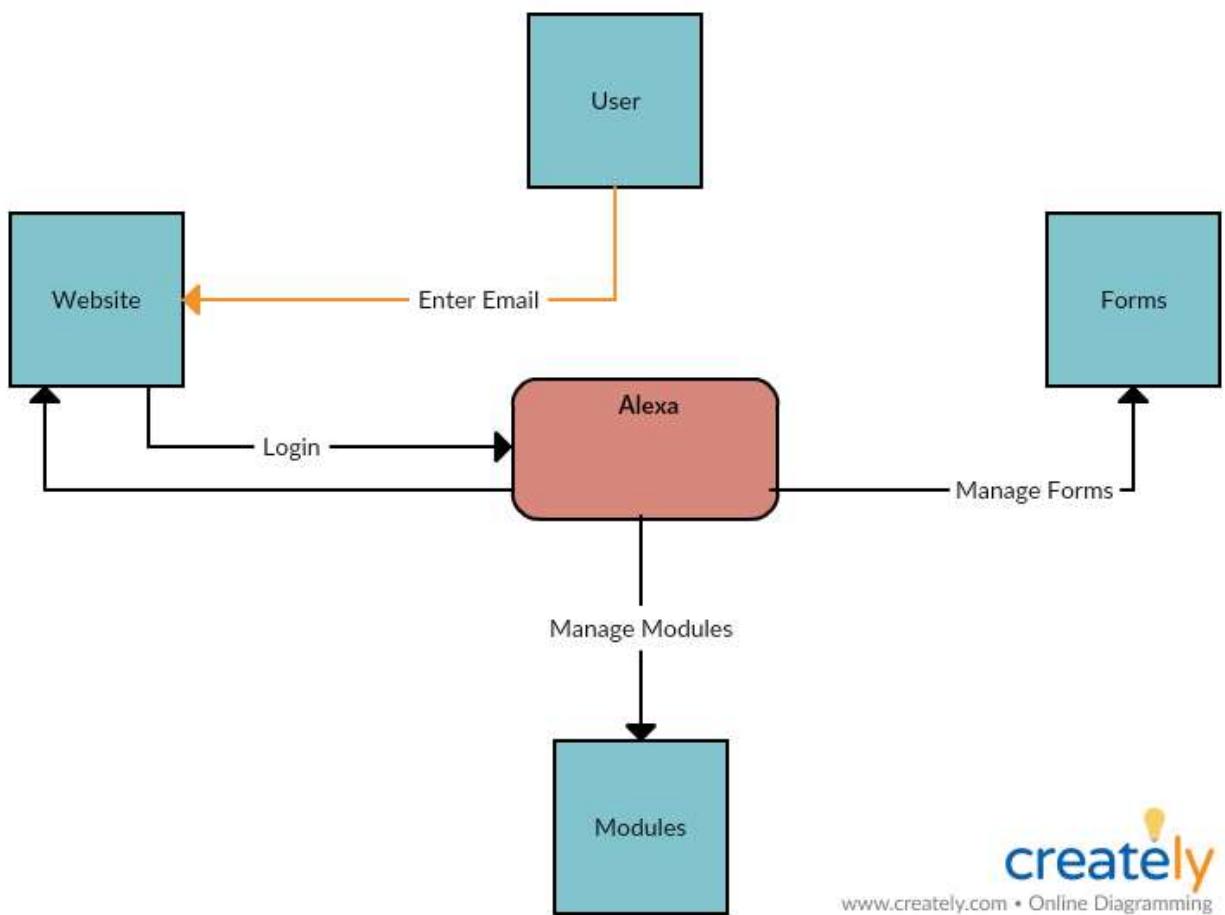


Fig 7: DFD lvl 0

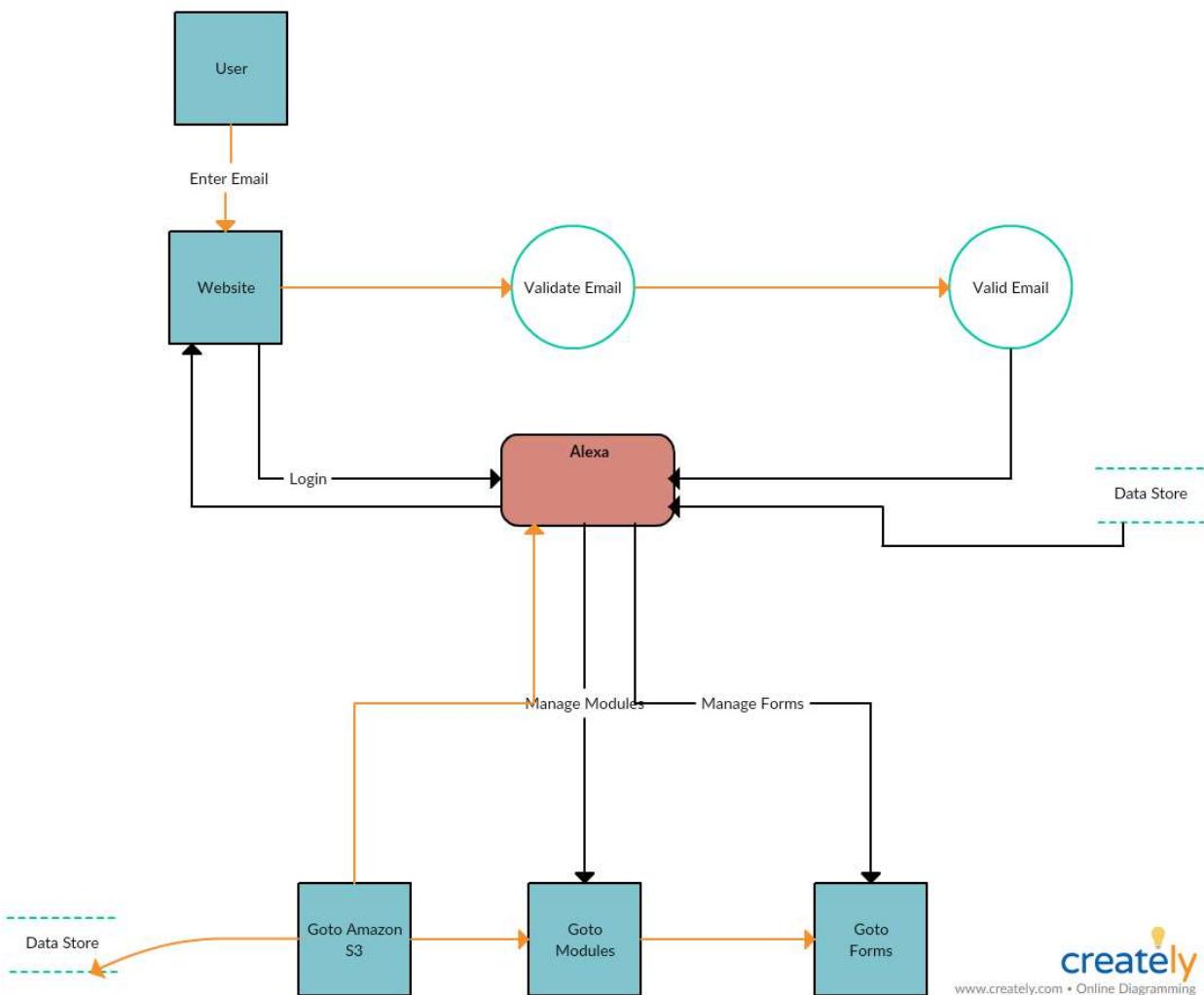


Fig 8: DFD level 1

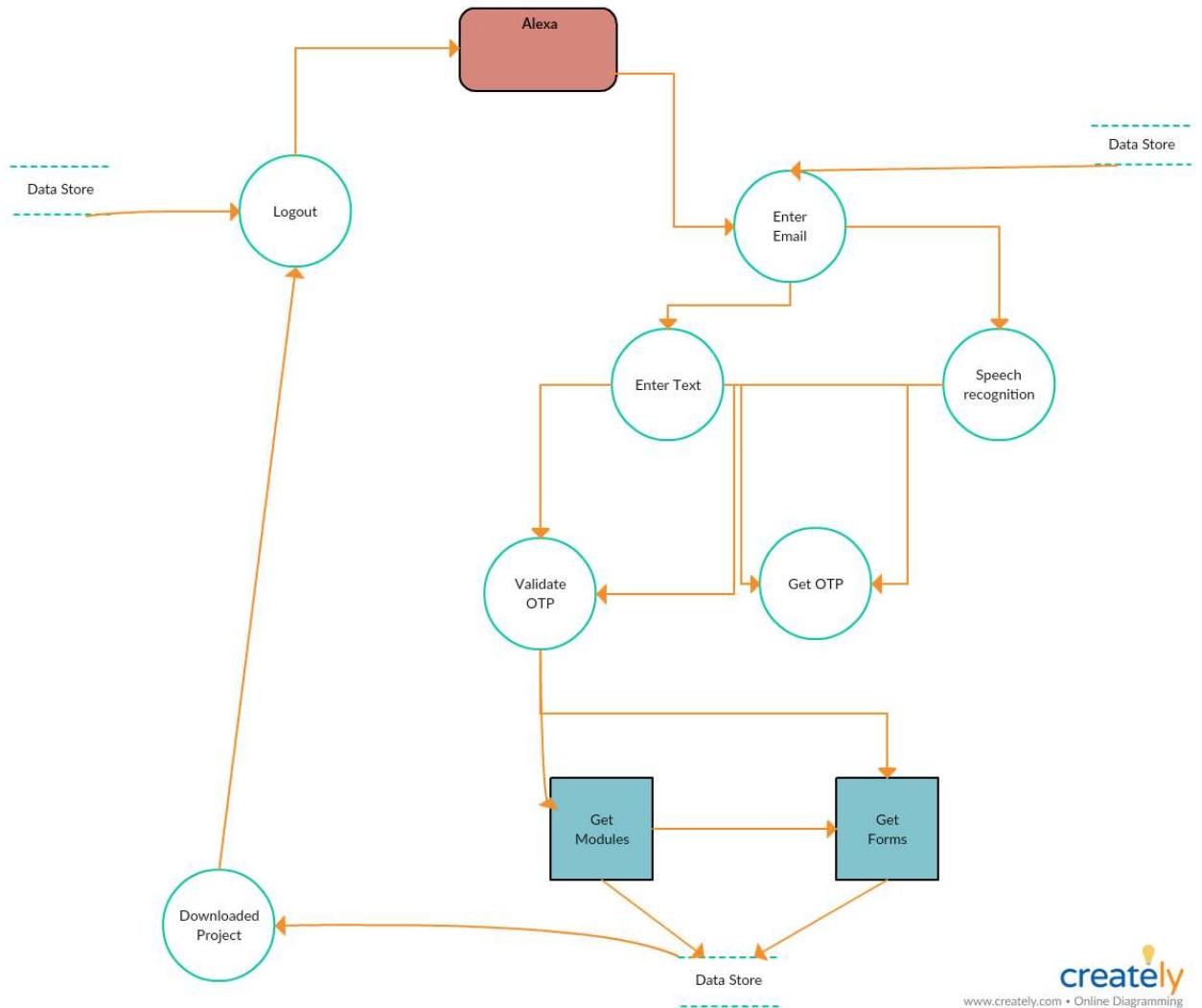


Fig 9: DFD level 2

5 System Design:

5.1 System Application Design:

5.1.1 Method pseudo code

When the user start the application for creation of form

First of all they want to enter the email-address

That email-address is verified with the registered email-address .

If the entered and registered email-address not matched

Then it will ask you to enter the valid email-address or email-address which is used for registration

If the email-address verified then otp is sent to the registered email-address

After that the user will ask what they want of any one of the operations (Crud)operations

C-CREATE

R-READ

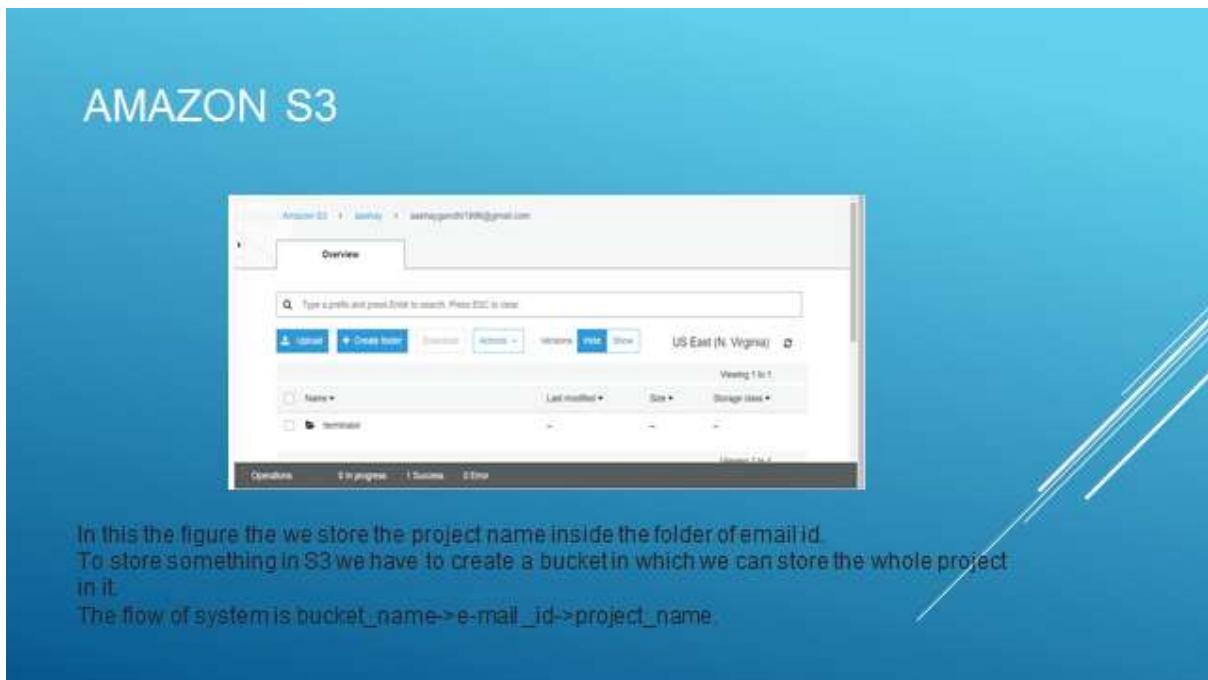
U-UPDATE

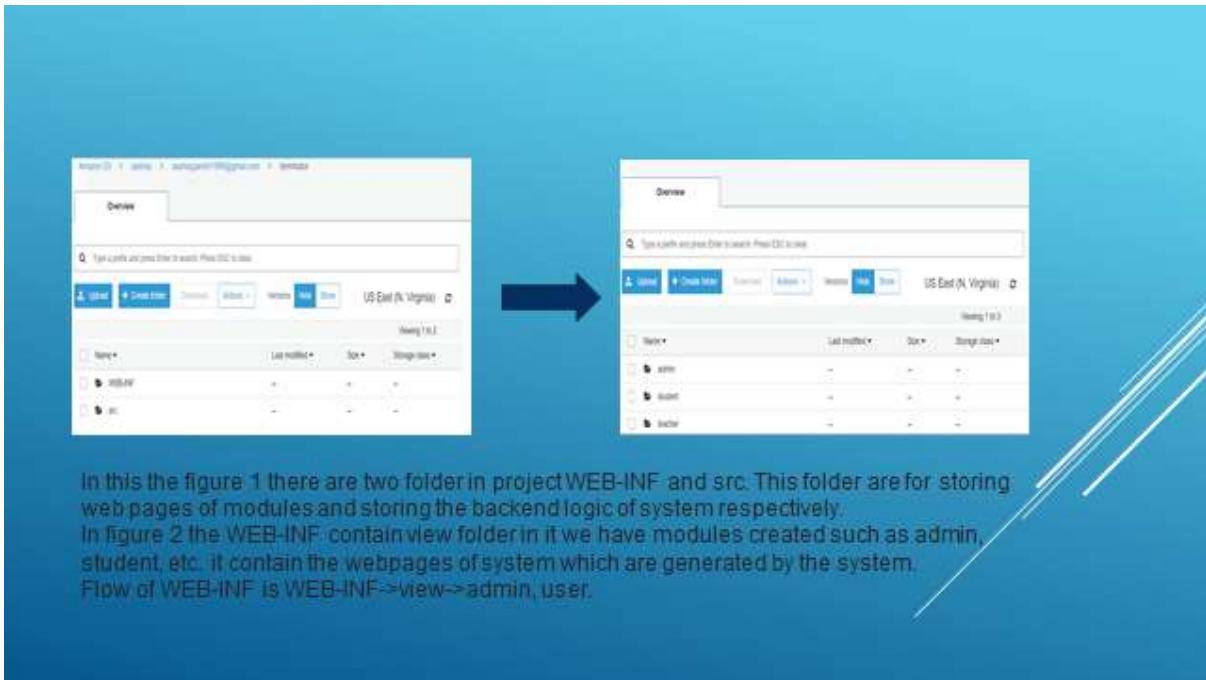
D-DELETE

User can create as many forms he want as per his convenience

5.2 Data Structure Design:

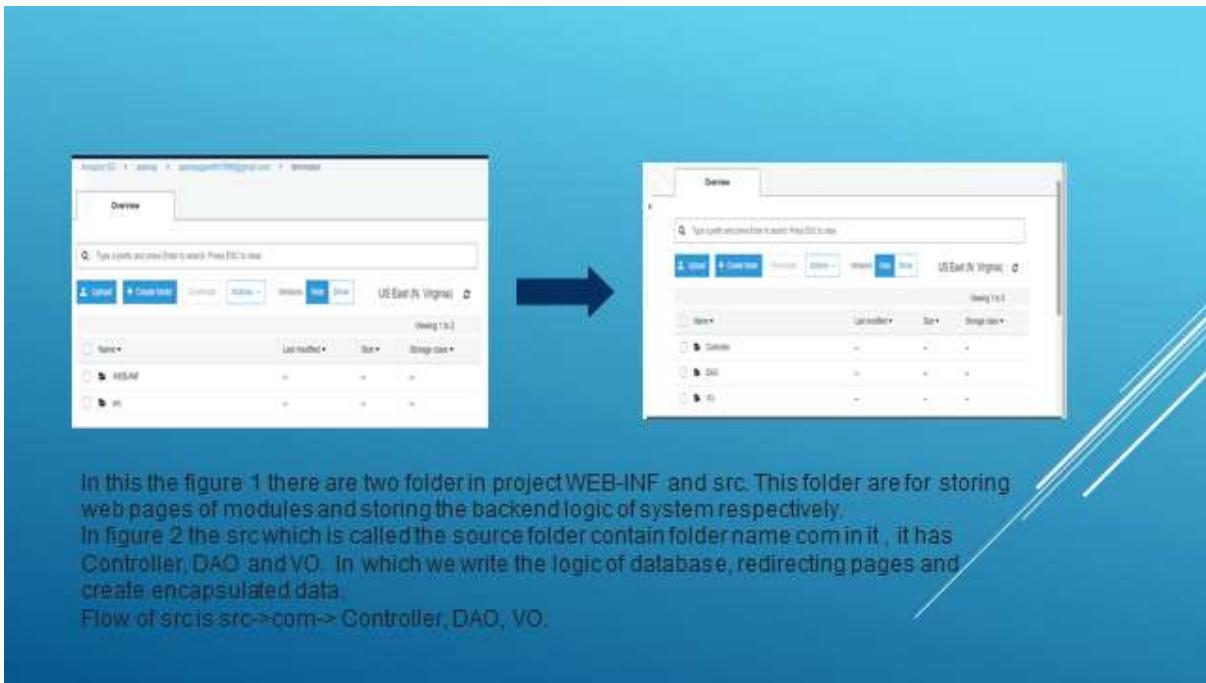
5.2.1 Table in Database:





In this the figure 1 there are two folder in project WEB-INF and src. This folder are for storing web-pages of modules and storing the backend logic of system respectively.

In figure 2 the WEB-INF contain view folder in it we have modules created such as admin, student, etc. it contain the webpages of system which are generated by the system.
Flow of WEB-INF is WEB-INF->view->admin, user.



In this the figure 1 there are two folder in project WEB-INF and src. This folder are for storing web-pages of modules and storing the backend logic of system respectively.

In figure 2 the src which is called the source folder contain folder name com in it , it has Controller, DAO and VO. In which we write the logic of database, redirecting pages and create encapsulated data.

Flow of src is src->com-> Controller, DAO, VO.

Amazon simple storage service :It is the storage for the internet.You can use Amazon s3 to store and retrieve any amount of data at any time,from anywhere on the web.You can complete this task by using Aws Management Console,Which is simple and intuitive web interface

Amazon S3 stores data as objects within buckets.An object consists of a file and optionally any metadata that describes that file.

To store an object in Amazon S3, we upload the file that we want to store in a bucket. When we upload a file we can set permissions on objects and on metadata.

5.3 Amazon Lamda



To run the Alexa we first need to upload the code in the Amazon Lambda through which we run the code which does is server less computation of the system.

In figure the "Email" is function name and the "Alexa skill set" is a trigger which is used to tell where we want the code to run.

The function code is uploaded in two ways through s3 or by uploading through pc and select the runtime environment for the code which can be any java 7, java 8, python 3 etc.

AWS LAMBDA:- It is a service which computes the code without any server. It is said to be serverless compute. The code is executed based on the response of events in AWS services such as adding/removing files in S3 Bucket, updating Amazon Dynamo DB tables, HTTP request from Amazon API Gateway etc.

Aws Lamda supports languages like NODEJS, Java, Python, C# and GO

5.1.1 Method pseudo code

When the user start the application for creation of form

First of all they want to enter the email-address

That email-address is verified with the registered email-address .

If the entered and registered email-address not matched

Then it will ask you to enter the valid email-address or email-address which is used for registration

If the email-address verified then otp is sent to the registered email-address

After that the user will ask what they want of any one of the operations (Crud)operations

C-CREATE

R-READ

U-UPDATE

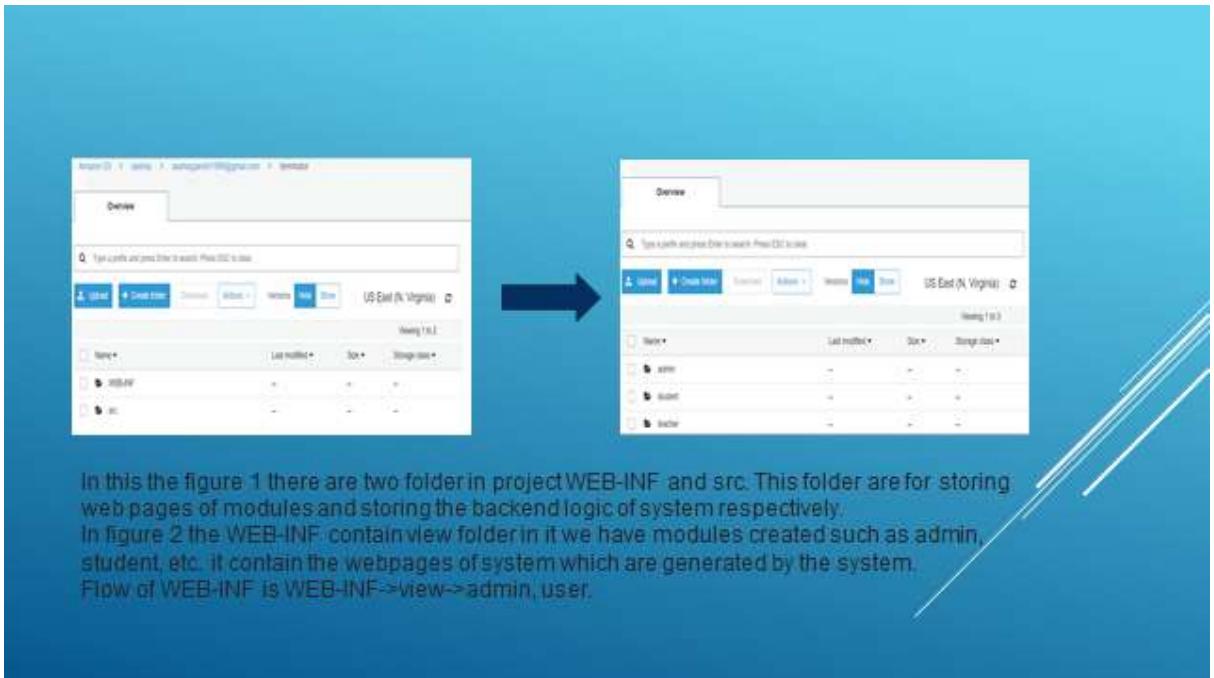
D-DELETE

User can create as many forms he want as per his convenience

5.2 Data Structure Design:

5.2.1 Table in Database:

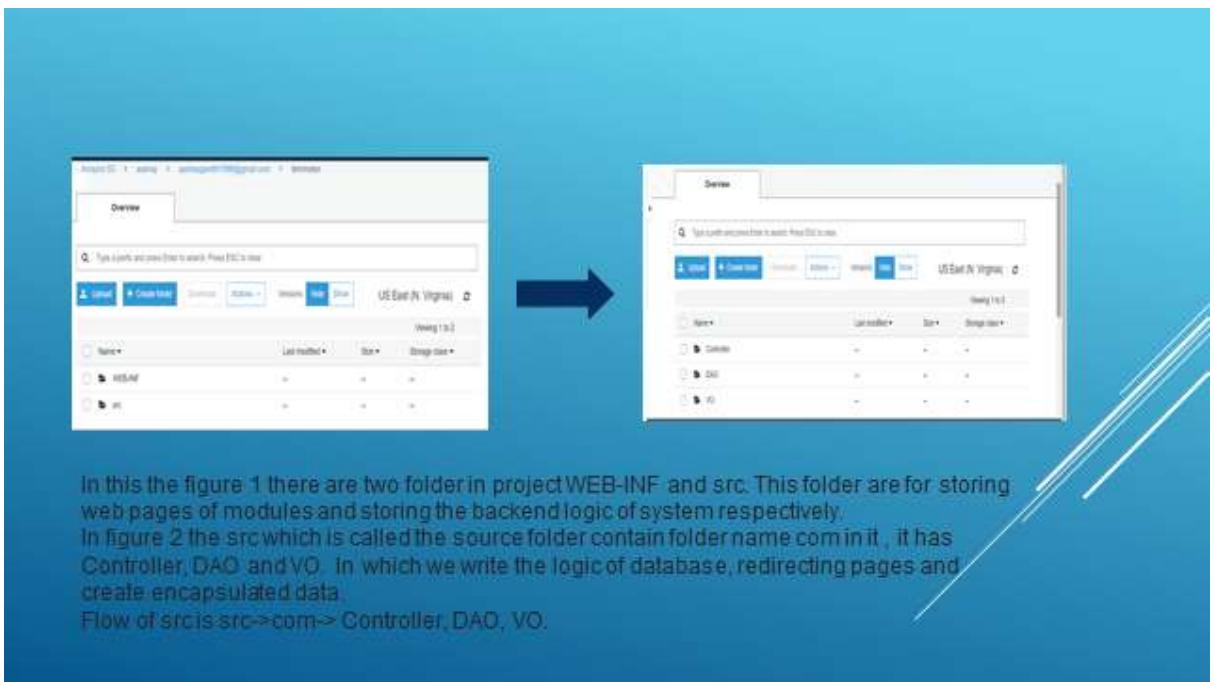




In this the figure 1 there are two folder in project WEB-INF and src. This folder are for storing web-pages of modules and storing the backend logic of system respectively.

In figure 2 the WEB-INF contain view folder in it we have modules created such as admin, student, etc. it contain the webpages of system which are generated by the system.

Flow of WEB-INF is WEB-INF->view->admin, user.



In this the figure 1 there are two folder in project WEB-INF and src. This folder are for storing web-pages of modules and storing the backend logic of system respectively.

In figure 2 the src which is called the source folder contain folder name com in it , it has Controller, DAO and VO. In which we write the logic of database, redirecting pages and create encapsulated data.

Flow of src is src->com-> Controller, DAO, VO.

Amazon simple storage service :It is the storage for the internet.You can use Amazon s3 to store and retrieve any amount of data at any time,from anywhere on the web.You can complete this task by using Aws Management Console,Which is simple and intuitive web interface

Amazon S3 stores data as objects within buckets.An object consists of a file and optionally any metadata that describes that file.

To store an object in Amazon S3, we upload the file that we want to store in a bucket. When we upload a file we can set permissions on objects and on metadata.

5.3 Amazon Lamda



To run the Alexa we first need to upload the code in the Amazon Lambda through which we run the code which does is server less computation of the system.

In figure the "Email" is function name and the "Alexa skill set" is a trigger which is used to tell where we want the code to run.

The function code is uploaded in two ways through s3 or by uploading through pc and select the runtime environment for the code which can be any java 7, java 8, python 3 etc.

AWS LAMBDA:- It is a service which computes the code without any server. It is said to be serverless compute. The code is executed based on the response of events in AWS services such as adding/removing files in S3 Bucket, updating Amazon Dynamo DB tables, HTTP request from Amazon API Gateway etc.

Aws Lamda supports languages like NODEJS, Java, Python, C# and GO

6 TESTING SPECIFICATION:

6.1 Purpose:

Testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs. Software testing can also be stated as the process of validating and verifying that a software program or application or product:

1. Meets the business and technical requirements that guided its design and development;
2. Works as expected;
3. Can be implemented with the same characteristics

6.2 Scope of Testing

Software testing, depending on the testing method employed, can be implemented at any time in the development process. However, most of the test effort occurs after the requirements have been defined and the coding process has been completed. As such, the methodology of the test is governed by the software development methodology adopted. Different software development models will focus the test effort at different points in the development process. Newer development models, such as Agile, often employ test driven development and place an increased portion of the testing in the hands of the developer, before it reaches a formal team of testers. In a more traditional model, most of the test execution occurs after the requirements have been defined and the coding process has been completed.

6.3 Test Plan

To test this application we are going with proper sequencing of testing like unit, integration, validation, GUI, Low level and High level test cases, major scenarios likewise. We will go with the GUI testing first and then integration testing. After integration testing performs the high level test cases and major scenarios which can affect the working on the application. We will perform the testing on the data transmitted using the various inputs and outputs and validate the results. It also intends to cover any

deviations that the project might take from the initially agreed Test Strategy in terms of scope, testing methodology, tools, etc. This test plan covers details of testing activities for this project and scope.

6.4 Test cases:

System Test case

Test case Id	Test case	Test case I/P	Actual Result	Expected Result	Test case criteria(P/F)
001	Login Functionality	Correct/Incorrect Name	Login Successful/Unsuccessful	Login Successful/Unsuccessful	P/F
002	Check response on entering VALID Agent Name and password	User name and Password	Reply with welcome message	Reply with welcome message	P
003	Check response on entering INVALID Agent Name and password	User name and Password	Reply with error message	Reply with error message	P
004	If code is generated	project name, form name, field type and name	Send Mail with link	Send Mail with link	P
005	Messaging from another email (not stored number in database)	MYHELP 1 N1	DON'T reply with anything	Not to Reply	p

Fig d. Test Case Table

7 Conclusion:

We propose a framework for the development of the application that will enable a person to make a CRUD operation Web pages written in javascript.

It uses Alexa to get input from the user such as project name, modules and forms, then generate the code as per the input and store the code in AmazonS3 then the code generated is mailed to the user on their registered email. Then the user can run the code on his system in Eclipse IDE by using his database. Everything was totally done in Java Script and within the time frame, our proposed framework for now can only provide CRUD functions.

Future Possibility:

We initially wanted to develop an application that including above functions can work with images and other type of login pages, this could have been done as the Alexa allow the multimedia messages for sending, But we proposed to develop this to initially so we didn't developed the Dynamically creating facility but in future after developing this framework we will include the Dynamically creating in our application.

References:

1. <https://www.alexa.com/>
2. <https://www.udemy.com/>
3. <https://www.tutorialspoint.com/tutorialslibrary.htm>
4. <https://www.eclipse.org/eclipseide/>

BIBLIOGRAPHY

Web Resources :

- <https://firebase.google.com/>
- <https://developer.amazon.com/studio/>
- https://www.tutorialspoint.com/android/android_studio
- www.udemy.com