HANDWRITTEN RECOGNITION USING NEURAL NETWORK

(Course Code: CA6CRP08)

Submitted in a partial fulfilment of the requirement for the award of the degree of

BACHELOR OF COMPUTER APPLICATION

Submitted by

REVATHI S KUMAR

200021092013

Under the guidance of

MS. ANU JOSEPH

DEPARTMENT OF COMPUTER APPLICATIONS

KRISTU JYOTI COLLEGE OF MANAGEMENT AND TECHNOLOGY,

CHANGANACHERRY

MARCH -2023

KRISTU JYOTI COLLEGE OF MANAGEMENT AND TECHNOLOGY

CHANGANACHERRY

DEPARTMENT OF COMPUTER APPLICATION

CERTIFICATE

Certify that the project record entitled "**Handwritten Recognition Using Neural Network** " is a bonafide report of the project done by **Revathi S Kumar** (**200021092013**) under our guidance and supervision is submitted in partial fulfilment of the Bachelor of Computer Applications, awarded by Mahatma Gandhi University, Kerala and that no part of this work has been submitted earlier for the award of any other degree.

Rev.Fr. Joshy Cheeramkuzhy CMI   Ms. Aby Rose Varghese

(Principal)   (Project Coordinator)

Department Seal

Mr. Roji. Thomas   Ms. Anu Joseph

(H.O.D)   (Project Guide)

Submitted for Project Evaluation and viva-voce held on........................Evaluator(s)

1.   2.

DECLARATION

I hereby declare that the project work entitled "**HANDWRITTEN RECOGNITION USING NEURAL NETWORK** " submitted to Mahatma Gandhi University in partial fulfilment of requirement for the award of degree of Bachelor of Computer Application from **Kristu Jyoti College of Management and Technology** , Changanacherry is a record of bonafide work done under the guidance of **Ms. Anu Joseph Assistant professor**, **Mr. Roji Thomas HOD** , Department of Computer Application , **Kristu Jyoti College of Management and Technology , Changanacherry** . This project has not been submitted in partial or fulfilment of any other degree/diploma /fellowship or similar of this university or any other university.

Place: Changanacherry   Revathi S Kumar

Date :   (200021092013)

ACKNOWLEDGEMENT

First of all, I thank God Almighty for being able to complete this project with success.

Then the college management who has been with me and provide most helpful facilities throughout the completion of the project. I am greatly indebted to Rev.Fr.Joshy Cheeramkuzhy CMI, Principal, Kristu Jyoti College, who wholeheartedly have me the permission and whose advice was a real encouragement.

I wish to record my deep sense of gratitude to Mr. Roji Thomas, HOD, Department of Computer Applications, for the constant supervision as well as providing necessary information regarding the project.

I would like to express my gratitude towards Ms. Aby Rose Varghese, Project Co-ordinator, for the valuable guidance, discussion and helpful counselling.

I am deeply indebted to Ms. Anu Joseph, Project Guide, for the valuable suggestions provided which helped me to overcome the major problems encountered during the completion of this project.

Last but least, I wish to express my gratitude and heartfelt thanks to my friends, family and whom with their valuable advice, encouragement and support for the successful completion of the project.

Revathi S Kumar

COMPANY CERTIFICATE

ABSTRACT

Handwritten Character recognition is the ability of a computer to receive and interpret intelligible handwritten input. Handwritten recognition is an emerging technology in image processing. Handwritten recognition is the wonders of empowering a machine to automatically realize the characters written in a user dialect. Character extraction has turned out to be a standout amongst the best candidates of innovation in the field of pattern recognition and artificial intelligence. In proposed technique, we present a handwritten character and digit recognition system based on different Deep learning technique. Handwritten character recognition plays an important role and is currently getting the attention of researchers because of possible applications in assisting technology for blind and visually impaired users, human–robot interaction, automatic data entry for business documents, etc. In the proposed work, we propose a technique to recognize handwritten characters and digits using deep learning approaches like Convolutional Neural Network (CNN) With, Adaptive Moment (Adam) Estimation and Dense Neural Networks. The proposed system has been trained on samples of large set of database images and tested on samples images from user defines data set and from this experiment we achieved very high recognition results. Experimental results are compared with another neural network-based algorithm.

INDEX

	• Economic feasibility e) Software engineering paradigm applied	
3.	Chapter 3: System Design a) Introduction b) Database Design • ER Model c) Process Design-Data flow d) Object oriented design UML • Activity Diagram • Sequence Diagram • Use case Diagram e) Modular Design • Modules Description f) Input Design g) Output Design	
4.	Chapter 4: System Environment a) Introduction b) Software requirement specification c) Hardware requirement specification	

	d) Tools , platform • Front end tool • Back end tool • Operating system	
5.	Chapter5: System Implementation a) Introduction b) Coding • Coding standards • Sample codes • Code validation & optimization c) Debugging d) Unit testing • Test plan &Test cases	
6.	Chapter 6: System Testing a) Introduction b) Integration testing c) System Testing • Test plan & Test cases	
7.	Chapter 7 : System Maintenance a) Introduction b) Maintenance	
	Chapter 8: System Security	

8.	Measures a) introduction b) operating system level security c) Database level security d) System level security	
9.	Chapter 9: System planning and scheduling a) Introduction b) Planning a software project • Steps involved in planning c) GANNT chart d) PERT chart	
10.	Chapter 10: System cost Estimation a) Introduction b) LOC Based Estimation	
	Chapter 11: Future Enhancement and Scope of Further Development. a) Introduction	

11.	b) Merits of the system c) Limitations of the system d) Future enhancement of the system	
12.	Chapter 12: Conclusion	
13.	Chapter 13: Bibliography	
14.	Chapter 14: Appendix • Screenshots	

CHAPTER 1 INTRODUCTION

INTRODUCTION

Text recognition is a territory of example distinguishing proof that has been the subject of extensive amid the current decades. Manually written text shows wide complex varieties. Penmanship is a standout amongst the most important mean in day-by-day discussion. Amid the current years, by far most of the conspicuous field of study and applications joined for bank check taking care of, sent wraps address perusing, and composed by hand message recognizable proof in records and recordings. Character Recognition System (CRS)used to distinguish mortal print image. These images might be alphabetic, numeric, punctuation and so forth. These images might be either printed or composed by submit an assortment of various size and textual style. All the more exactly character recognition is the way toward distinguishing and perceiving character from information picture and change it into American Standard Code for Information Interchange or other comparing machine editable form. The chore of recognition comprehensively segregates into two kinds: written by hand and machine printed. The printed character reference is uniform and extraordinary.

Deep learning Techniques have been effectively applied to various areas like image classification, speech recognition, Medical Images detection, face detection, satellite images, recognizing traffic signs and pedestrian detection and so on. The outcome of deep learning

techniques is also prominent, and in some cases the results are superior to human experts in the past years. From last few years most of the problems are also being re-experimented with deep learning techniques with the view to achieving improvements in the existing findings. Different architectures of deep learning have been introduced in recent years, such as convolutional neural networks, deep networks, and recurrent neural networks. The entire architecture has shown the expertise in different areas. Character recognition is one of the areas where machine learning techniques have been extensively experimented. The first deep learning technique, which is one of the leading machine learning techniques, was proposed for character recognition in 1998 on MNIST database. The deep learning techniques are basically composed of multiple hidden layers, and each hidden layer consists of multiple neurons, which compute the suitable weights for the deep network. A lot of computing power is needed to compute these weights, and a powerful system was needed, which was not easily available at that time. Since then, the researchers have drawn their attention to finding the technique which needs less power by converting the images into feature vectors. In the last few decades, a lot of feature extraction technique have been proposed such as HOG (histogram of oriented gradients) and many others techniques are used as prominent feature extraction methods, which have been experimented for many problems like image recognition, character recognition, face detection, etc. Feature extraction is one type of dimensionality reduction technique that represents the important parts of a large image into a feature vector. These features are handcrafted and clearly designed by the research community. The robustness and performance of these features depend on the skill and the knowledge of each researcher. There are the cases where some vital features may be unseen by the researchers while extracting the features from the image and this may result in a high classification error. Deep learning inverts the process of handcrafting and designing features for a particular problem into an automatic process to compute the best features for that problem. A convolutional neural network has multiple convolutional layers to extract the features automatically. The features are extracted only once in most of the shallow learning models, but in the case of deep learning models, multiple convolutional layers have been adopted to extract discriminating features multiple times. This is one of the reasons that deep learning models are generally successful. And also, in Deep feed forward neural networks the features are compute automatically by using different number of hidden layers in it. Owing to their great success, many leading companies have also introduced deep models. Character recognition is a field of image processing where the image is recognized and converted into a machine-readable format. As discussed above, the deep learning techniques and especially convolutional neural networks have been used for image detection and recognition. It has also been successfully applied on Roman (MNIST), Chinese, Bangla and Arabic languages. In this work, a convolutional neural network is applied for handwritten English characters and digits recognition.

PROBLEM STATEMENT

 The handwritten digit recognition is the capability of computer applications to recognize the human handwritten digits. It is a hard task for the machine because handwritten digits are not perfect and can be made with many different shapes and sizes. The handwritten digit recognition system is a way to tackle this problem which uses the image of a digit and recognizes

the digit present in the image. Convolutional Neural Network model created using PyTorch library over the MNIST dataset to recognize handwritten digits .

Handwritten Digit Recognition is the capability of a computer
to fete the mortal handwritten integers from different sources like images, papers, touch defenses , etc,and classify. This has been a Content of bottomless- exploration in
the field of deep literacy. Number recognition has numerous operations like number plate recognition, postal correspondence sorting, bank check processing, etc.

Handwritten number recognition,we face numerous challenges because
of different styles of jotting of different peoples as it is not an Optic character recognition. This exploration provides a comprehensive comparison between different machine literacy and deep literacy for the purpose of handwritten number recognition. Vector Machine, Multilayer Perceptron, and Convolutional Neural Network. The comparison between these algorithms is carried out on the base of their delicacy, crimes, and testing
training time corroborated by plots and maps that have been constructed using matplotlib for visualization.

SCOPE AND RELEVANCE OF THE PROJECT

- Electronic form filling

- Writing electronic applications in one's own handwriting and native script

- Automated music symbol notation reader

- Alternative to hardware and software keyboards

- Putting in the mathematical equations by simple handwriting

- Writing and sending SMS in his/her mother tongue

- Biometrics and forensics

- Digitization of palm leaf manuscripts

- Automatic conversion of prescription to typed form

The artificial neural networks approach is considered as the best way to develop systems for recognizing handwriting. Neural networks help to simulate how the human brain works when reading handwriting in a more simplified form. It allows machines to match and even exceed human capabilities at reading handwriting.

OBJECTIVES

Handwritten Character recognition is the ability of a computer to receive and interpret intelligible handwritten input. Handwritten recognition is an emerging technology in image processing. Handwritten recognition is the wonders of empowering a machine to automatically realize the characters written in a user dialect. Character extraction has turned out to be a standout amongst the best candidates of innovation in the field of pattern recognition and artificial intelligence. In proposed technique, we present a handwritten character and digit recognition system based on different Deep learning technique. Handwritten character recognition plays an important role and is currently getting the attention of researchers because of possible applications in assisting technology for blind and visually impaired users, human–robot interaction, automatic data entry for business documents, etc. In the proposed work, we propose a technique to recognize handwritten characters and digits using deep learning approaches like Convolutional Neural Network (CNN) With, Adaptive Moment (Adam) Estimation and Dense Neural Networks. The proposed system has been trained on samples of large set of database images and tested on samples images from user defines data set and from this experiment we achieved very high recognition results. Experimental results are compared with another neural network-based algorithm.

CHAPTER 2 SYSTEM ANALYSIS

EXISTING SYSTEM

U. Pal et al, have proposed a modified quadratic classifier-based scheme to recognize the offline handwritten numerals of six popular Indian scripts.

Multilayer perceptron has been used for recognizing Handwritten English characters. The features are extracted from Boundary tracing and their Fourier Descriptors.

Character is identified by analyzing its shape and comparing its features that distinguish each character.

Also, an analysis has been carried out to determine the number of hidden layer nodes to achieve high performance of back propagation network.

Dinesh et al have used horizontal/vertical strokes, and end points as the potential features for recognition and reported a recognition accuracy of 90.50% for handwritten Kannada numerals.

U. Pal et al have proposed zoning and directional chain code features and considered a feature vector of length 100 for handwritten numeral recognition and have reported a high level of recognition accuracy.

LIMITATIONS OF EXISTING SYSTEM

Existing techniques uses the thinning process which results in the loss of features. Feature extraction process is complex and time consuming.

PROPOSED SYSTEM

Proposed system approaches the problem of handwritten recognition of English alphabets. There are several steps proposed to achieve handwritten recognition with good accuracy. It is necessary to perform several document analysis operations prior to recognizing text. Common pre-processing operations carried out in the proposed techniques are thresholding, the task of converting a RGB to gray-scale image, noise removal Text identification is identified in the proposed system using Contour finding to apply Neural Network classifier to recognize characters.

ADVANTAGES OF PROPOSED SYSTEM

Handwriting recognition helps to transform the writings in the papers to a text document format which can also be said as readable electronic format.

FEASIBILITY STUDY

In feasibility study, we analyzes the feasibility of proposed system that is whether it satisfies all the necessary requirements. First is to study and analyzing the existing system, the next is to do the feasibility study for the project. Feasibility study includes consideration of all the possible ways to provide a solution to the existing system problems. The proposed solution should satisfy all the user requirements.

A feasibility analysis usually involves a through assignment of the operational, financial and technical aspects of a proposal. Feasibility study is the test of the system proposal made to identify whether the system will be cost effective from a business point of view and whether it can be developed with the given budgetary constraints. A feasibility study should be relatively cheap and done at the earliest possible time. Depending on the study, the decision is made whether to go ahead with a more detailed analysis.When a new project is proposed, informally goes through feasibility assignment. Feasibility study is carried out to determine whether the

proposed system is possible to develop with available resources and what should be the cost consideration. Facts considered in the feasibility analysis were-

- Technical Feasibility
- Economic Feasibility
- Operational Feasibility

Technical Feasibility

Technical feasibility deals with the hardware as well as software requirements.Technology is not a constraint to type system development. We have to find out whether a necessary technology, the proposed requirement have the capacity to hold the data, which is used in the project, should be checked to carry out this technical feasibility. In our system the technical feasibility is all the data's are to be in correct because of the automation of the system.

Economical Feasibility

Feasibility study presents the tangible and intangible benefits by comparing the development and operational cost. The technique of cost benefits analysis is often used as a basis for assessing economic feasibility. This system needs some more initial investment than the existing system, but it can be justifiable that it will improve quality of service.

Operational Feasibility

This analysis involves how it will work when it is installed and the assessment of political and managerial environment in which it is implemented. People are inherently resistant to change and computers have been known to change. The new proposed is very much useful to the user and therefore it will accept broad audience from around the world.

CHAPTER 3 SYSTEM DESIGN

INTRODUCTION

System designing in terms of software engineering has its own value and importance in the system development process as a whole. To mention it may though seem as simple as anything or simply the design of systems, but in a broader sense it implies a systematic and rigorous approach to design such a system which fulfils all the practical aspects including flexibility, efficiency and security. Systems design is the process of defining the architecture, components, modules, inter- faces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.

The system design covers the following.

• Reviews the current physical system.

• Prepares output specifications.

• Prepares input specifications.

• Prepares edit, security and control specifications.

• Specifies the implementation plan.

• Prepares a logical design walk through of the information flow, output, input, controls and implementation plan.

DATABASE DESIGN

In designing a database application, you must set up not only the program 's routines for maximum performance, but you must pay attention also to the physical layout of the data storage.

A good database design does the following:

• Provides minimum search times when locating specific records.

• Stores the data in the most efficient manner possible to keep the database from growing too large.

• Makes data updates as easy as possible.

• It is flexible enough to allow inclusion of new functions required of the program.

Normalization

It is a process of converting a relation to a standard form. The process is used to handle the problems that can arise due to data redundancy i.e., repetition of data in the database, maintain data integrity as well as handling problems that can arise due to insertion, updating, deletion anomalies.

Insertion anomaly: Inability to add data to the database due to absence of other data.

Deletion anomaly: Unintended loss of data due to deletion of other data. Update anomaly: Data incenses- tenkey resulting from data redundancy and partial update. Decomposing is the process of splitting relations into multiple relations to eliminate anomalies and maintain anomalies and maintain data integrity. To do this we use normal forms or rules for structuring relation. Normal Forms are the rules for structuring relations that eliminate anomalies.

First Normal Form:

A relation is said to be in first normal form if the values in the relation are atomic for every attribute in the relation. By this we mean simply that no attribute value can be a set of values or, as it is sometimes expressed, a repeating group.

Second Normal Form:

A relation is said to be in second Normal form is it is in first normal form and it should satisfy any one of the following rules.

• Primary key is a not a composite primary key.

• No non key attributes are present.

• Every non key attribute is fully functionally dependent on full set of primary keys.

Third Normal Form:

A relation is said to be in third normal form if their exits no transitive dependencies.

Transitive Dependency:

Self-two non-key attributes depend on each other as well as on the primary key then they are said to be transitively dependent.

The above normalization principles were applied to decompose the data in multiple tables thereby making the data to be maintained in a consistent state. The database is implemented using a DBMS package. Each particular DBMS has unique characteristics and general technique for database design. The application stores the information relevant for processing to SQL database. This SQL database contains tables where each table corresponding to one particular type of information. Each piece of information in a table is called a field for column. A table also contains records, which is a set of fields. These are primary key field that are uniquely identifying a record in a table. There are also fields that contain primary key form another table called foreign key.

Candidate key

In the relational model, a candidate key of a relational variable is a set of attributes of that relation variable such that at all times it holds in the relation assigned to that variable that there are no two distinct tuples with the same values for these attributes and there is not a proper subset of this set of attributes for which holds.

Primary key

In relational database design, a unique key to uniquely identify each row in a table. A uniquely key or primary key. comprises a single column or set of columns. No two distinct rows in a table can have the same value in those columns. Depending on its design, a table may have arbitrarily many unique keys but at most one primary key. A unique key must uniquely identify all possible rows that exist in a table and not only the currently existing rows.

Foreign key

In the context of relational databases, a foreign key is a referential constraint between two tables. The foreign key identifies a column or a set of columns in another table. The columns in the referencing table must be the primary key or other candidate key in the referenced table. The values in one row of the referencing columns must occur in a single row in the referenced table. Thus, a row in the referencing table cannot contain values that don 't exists in the referenced table.

ENTITY RELATIONSHIP DIAGRAM

The ER diagram is a graphical representation of entities and their relationships to each other, typically used in regard to the organization of data within database or information systems. An entity is a piece of data an object or concept about which data is stored. A relationship is now the data is shared between entities.
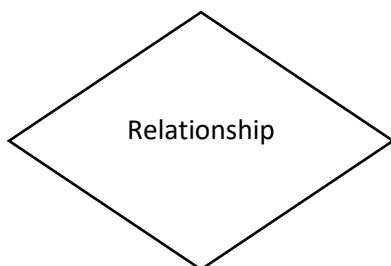
ER-SYMBOLS

ENTITY

An entity is defined as a thing which is recognized as being capable of an index- pendent existence and which can be uniquely identified. An entity is abstract from the complexities of a domain. When we speak of an entity, we normally speak of some aspect of real world which can be distinguished from other aspect of the world.
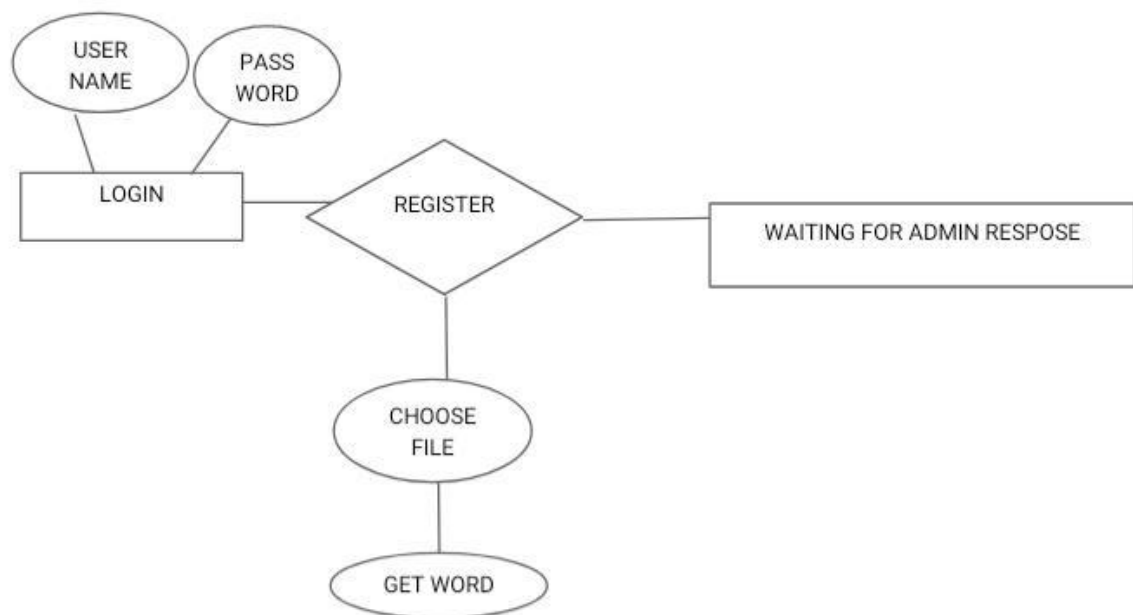
RELATIONSHIP

Relationship is meaningful association between or among entities. A relationship provides useful information that could not be discerned with just the entity types.

ATTRIBUTES
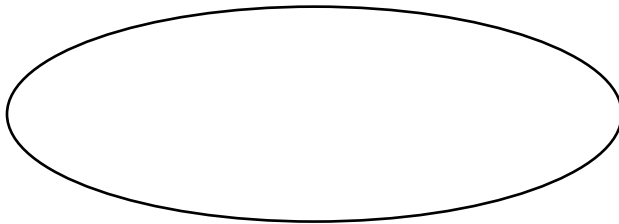
Attribute are characteristics of an entity, a many-to-one relationship, or one-to-one relationship.

ER DIAGRAM OF HANDWRITTEN RECOGITION USING NEURAL NETWORK.

DATA DICTIONARY

Data Dictionary is the major component in the structured analysis model of the system. It lists all the data items appearing in DFD. A data dictionary in Software Engineering means a file or a set of files that includes a database's metadata (hold records about other objects in the database), like data ownership, relationships of the data to another object, and some other data.

Case Tools is used to maintain data dictionary as it captures the data items appearing in a DFD automatically to generate the data dictionary.:

COMPONENTS:

• Name of the item: It can be your choice.

• Aliases: It represents another name.

• Description: Description of what the actual text is all about.

• Related data items: with other data items.

• Range of values: It will represent all possible answers.

NOTATIONS

The Notations used within the data dictionary as follows:

Notations	Meaning
X = a+b	X consists data elements a and b.
X = [a/b]	X consists of either elements a or b.
X = aX	X consists of optimal data elements.
X = y[a]	X consists of y or more events of data element a.
X = [a] z	X consists of z or less events of data element a.
X = y [a] z	X consists of some events of data elements between y and z.

FEATURES

• It helps in designing test cases and designing the software.

• It is very important for creating an order list from a subset of the items list.

• It is very important for creating an order list from a complete items list.

• The data dictionary is also important to find the specific data item object from the list.

USES:

• Used for creating the ordered list of data items.

• Used for creating the ordered list of a subset of the data items.

• Used for Designing and testing software in Software Engineering.

• Used for finding data items from a description in Software Engineering.

IMPORTANACE OF DATA DICTIONARY

• It provides developers with standard terminology for all data.

• It provides developers to use different terms to refer to the same data.

• It provides definitions for different data.

• Query handling is facilitated if a data dictionary is used in RDMS.

DATAFLOW DIAGRAM

A data-flow diagram (DFD) is a way of representing a flow of a data of a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart.

There are several notations for displaying data-flow diagrams. The notation pre- scented above was described in 1979 by Tom DeMarco as part of Structured Analysis.

For each data flow, at least one of the endpoints (source and / or destination) must exist in a process. The refined representation of a process can be done in another dataflow diagram, which subdivides this process into sub processes.

Data-flow diagrams can be regarded as inverted Petri nets, because places in such networks correspond to the semantics of data memories. Analogously, the semantics of transitions from Petri nets and data flows and functions from data-flow diagrams should be considered equivalent.
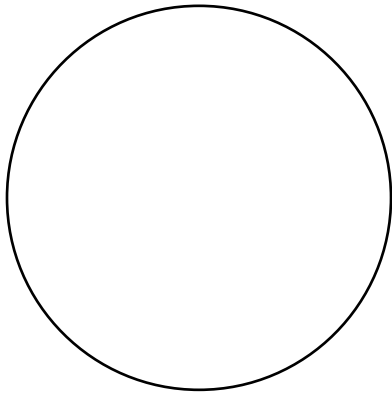
SYMBOLS USED IN DFD

RECTANGLE

An entity: A source of data or a destination for data.

CIRCLE

A process or task that is performed by the system.

OPEN RECTANGLE

A data store, a place where data is held between processes.

ARROW

It defines data flow. It is a pipeline through which information flows.

DFD HIERARCHY

To make the DFD more transparent (i.e., not too many processes), multi-level DFDs can be created. DFDs that are at a higher level are less detailed (aggregate more detailed DFD at lower levels). The contextual DFD is the highest in the hierarchy. The so-called zero level is followed by DFD 0, starting with process numbering (e.g., process 1, process 2). In the next, the so-called first level – DFD 1 – the numbering continues. E.g., process 1 is divided into the first three levels of the DFD, which are numbered 1.1, 1.2 and 1.3. Similarly, processes in the second level (DFD 2) are numbered egg 1.1.1, 1.1.2, 1.1.3 and 1.1.4.The number of levels depends on the size of the model system. DFD 0 processes may not have the same number of decomposition levels. DFD 0 contains the most important (aggregated) system functions. The lowest level should include processes that make it possible to create a process specification (Process Specification) for roughly one A4 page. If the mini-specification should be longer, it is appropriate to create an additional level for the process where it will be decomposed into multiple processes. For a clear overview of the entire DFD hierarchy, a vertical (cross- sectional) diagram can be created. The warehouse is displayed at the highest level where it is first used and at every lower level as well.

RULES FOR CREATING DFD

Entity names should be comprehensible without further comments. DFD is a system created by analysts based on interviews with system users. It is determined for system developers, on one hand, project contractor on the other, so the entity names should be adapted for model domain or amateur users or professionals. Entity names should be gen- earl (independent, e.g., specific individuals carrying out the activity), but should clearly specify the entity. Processes should be numbered for easier mapping and referral to spa- civic processes. The numbering is random; however, it is necessary to maintain consistency across all DFD levels (see DFD Hierarchy). DFD should be clear, as the maximum num- beer of processes in one DFD is recommended to be from 6 to 9, minimum is 3 processes in one DFD. The exception is the so-called contextual diagram where the only process symbolizes the model system and all terminators with which the system communicates.

DATA FLOW DIAGRAM

Level 0

Level 1-User
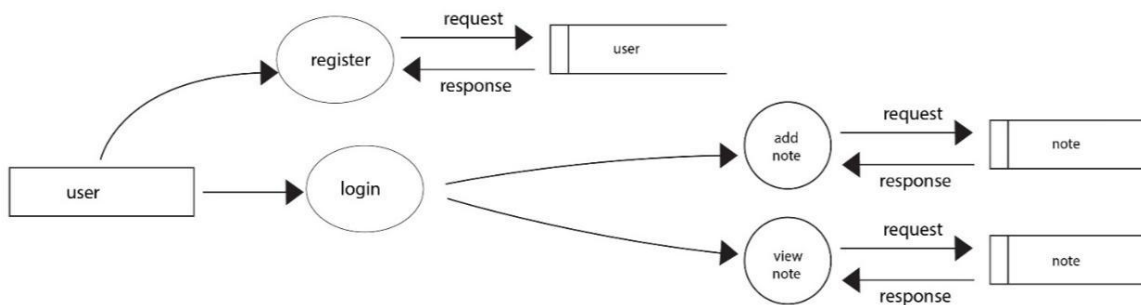
UML DIAGRAM

UML stands for Unified Modelling Language. It's a rich language to model software solutions, application structures, system behaviour and business processes. There are 14 UML diagram types to help you model these behaviours.
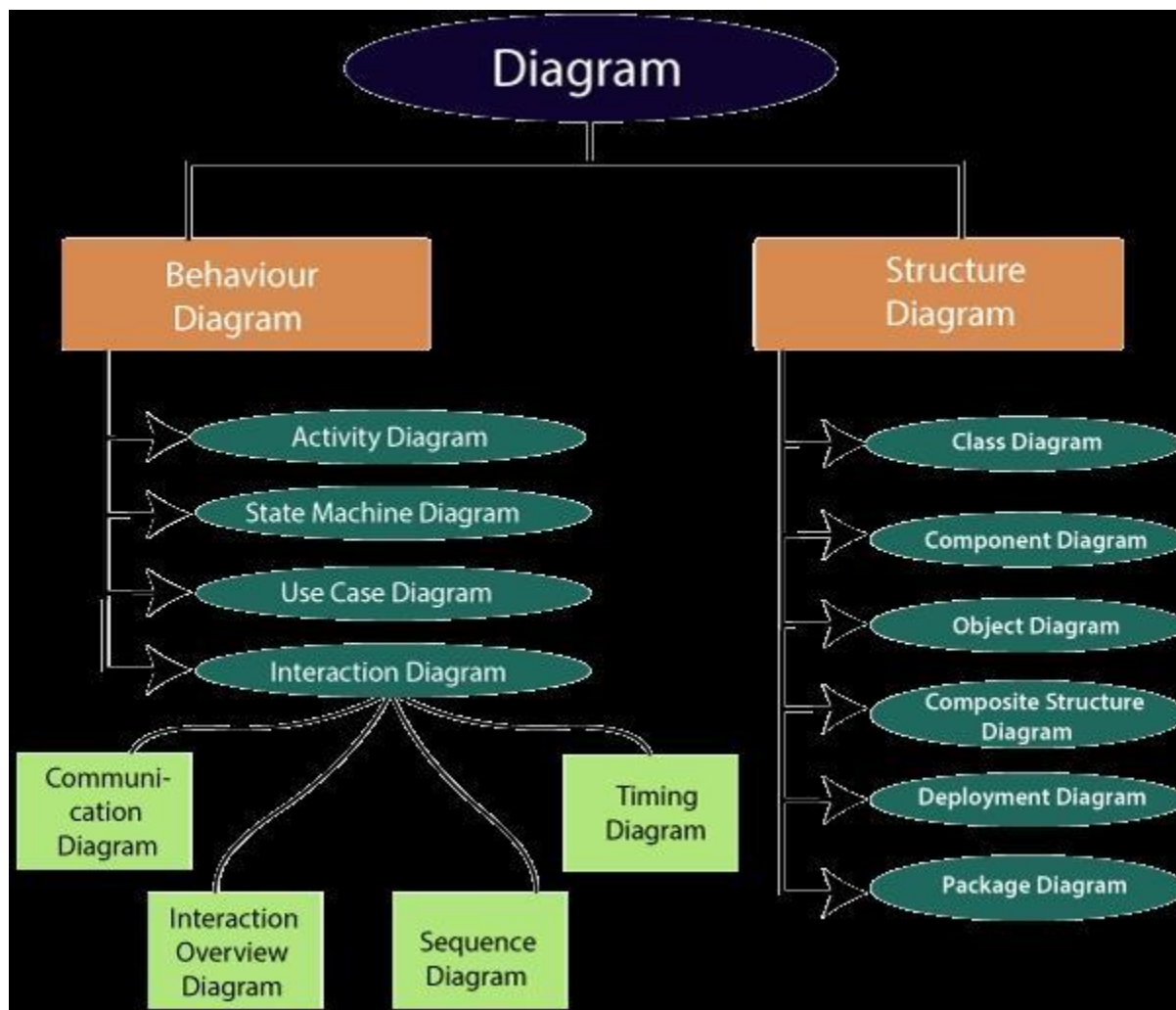
ACTIVITY DIAGRAM
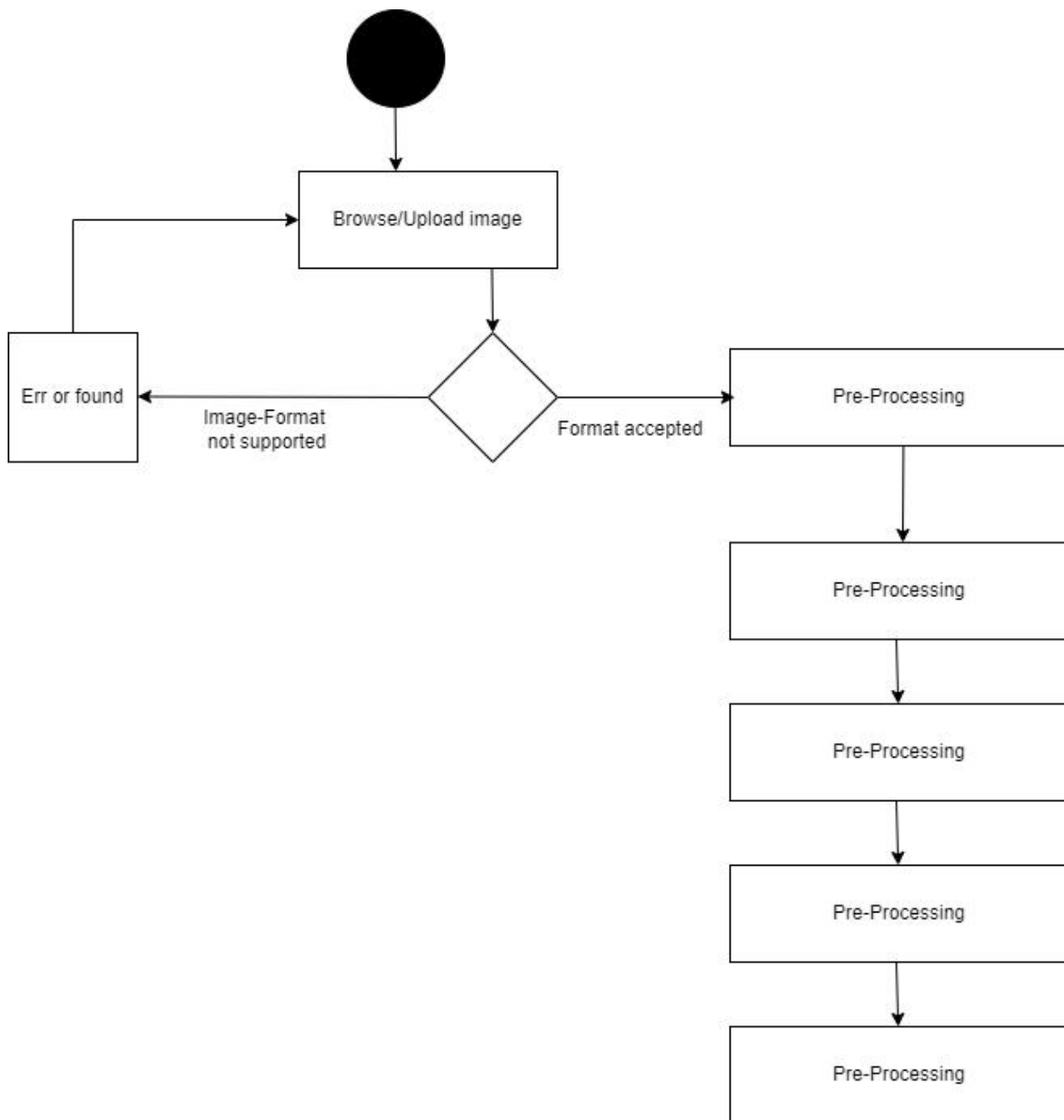
USECASE DIAGRAM

CLASS DIAGRAM

user

id	int
name	string
phoneno	varchar
email	varchar

login   ( )
register ( )

view note ( )
convert   ( )
addnote ( )

system

image varchar ( )

view note
convert ( )

transalate ( )
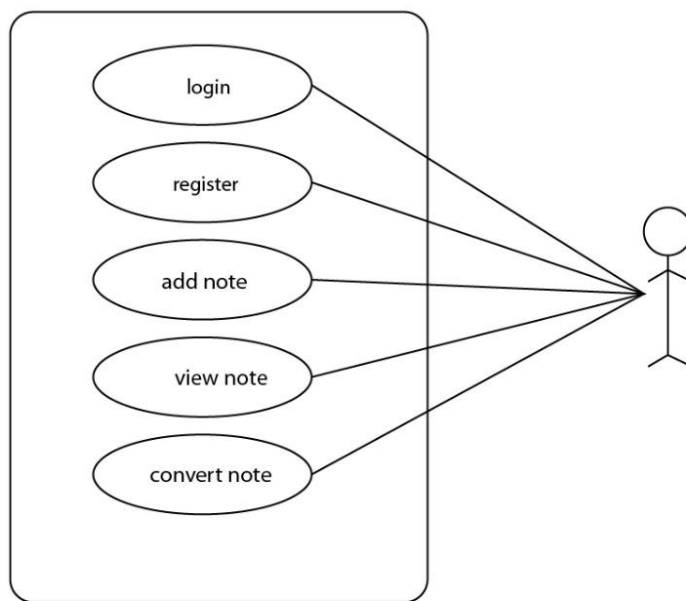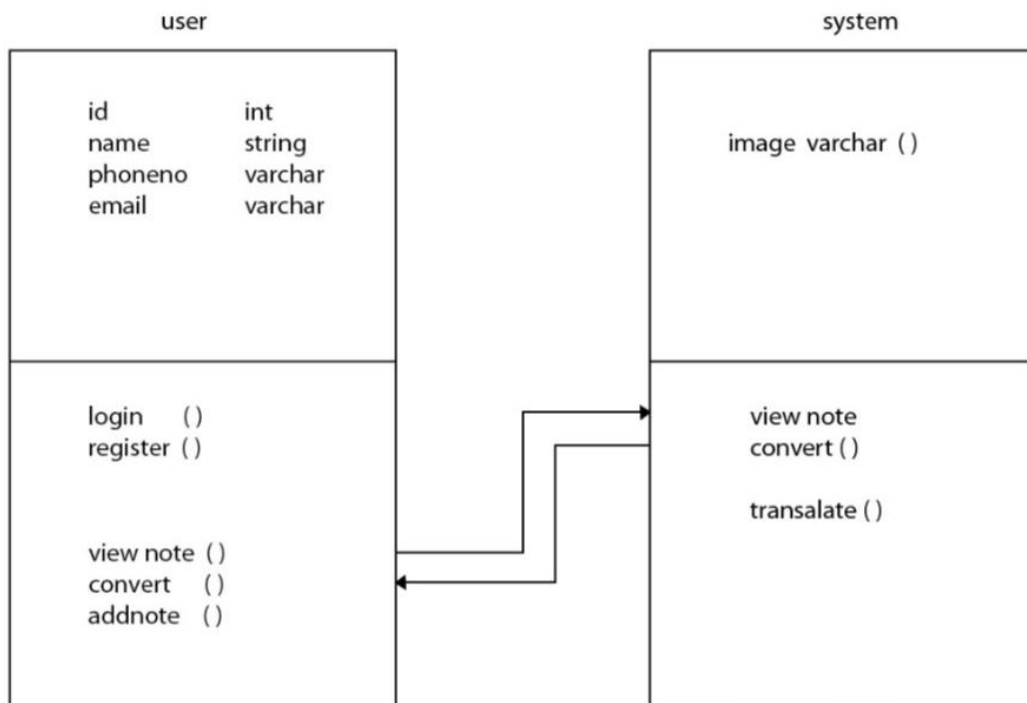
SEQUENCE DIAGRAM OF PREPROCESSING

Handwritten Recognition Using Neural Network

User

GUI

System

1.Display menu

2.Load/upload image

3.Process image

4.Pre processing

5.Segmentation

6.Recognition

7..Generator otp

8.Display o/p

9.Browse/upload resume

4.Error found

10.Process image

12.Send errors image

13.Display Errors
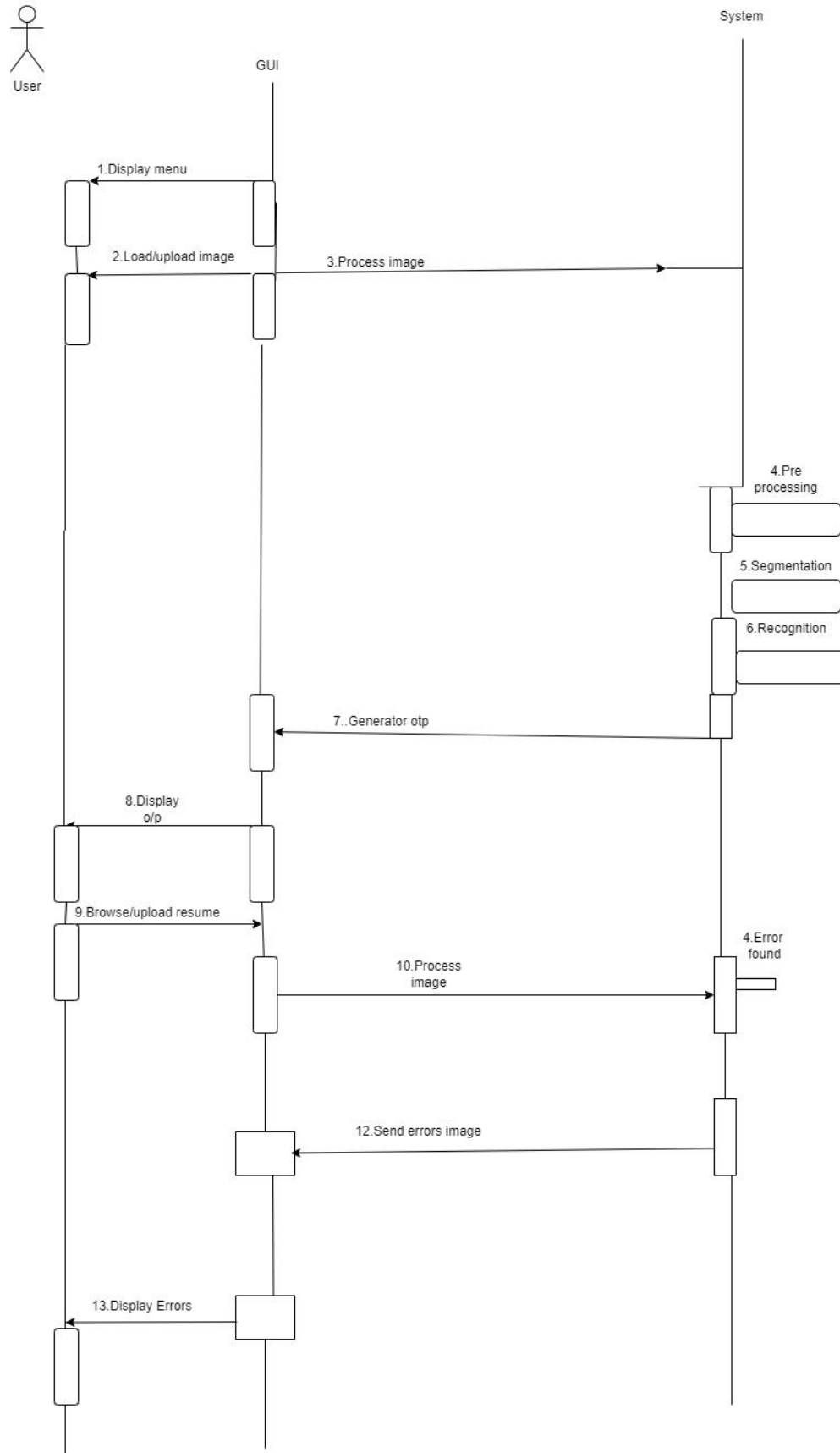
COMPONENT DIAGRAM
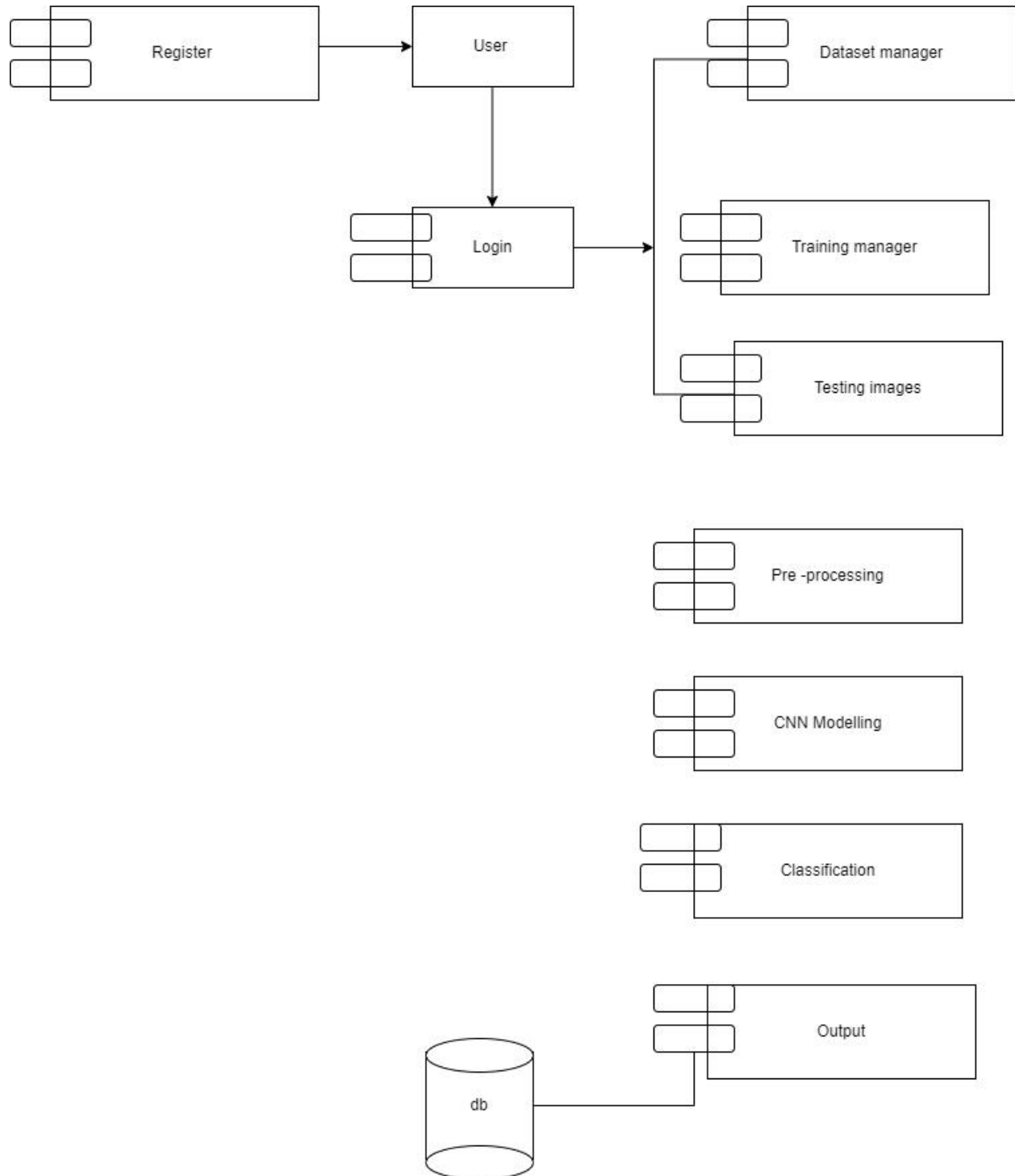
SEQUENCE DIAGRAM OF CLASSIFICATION

Handwritten Recognition Using Neural Network

u!User

S!System

Load Image()

Initial image()

Save image()

Apply grey scale()

thining()

Save image()
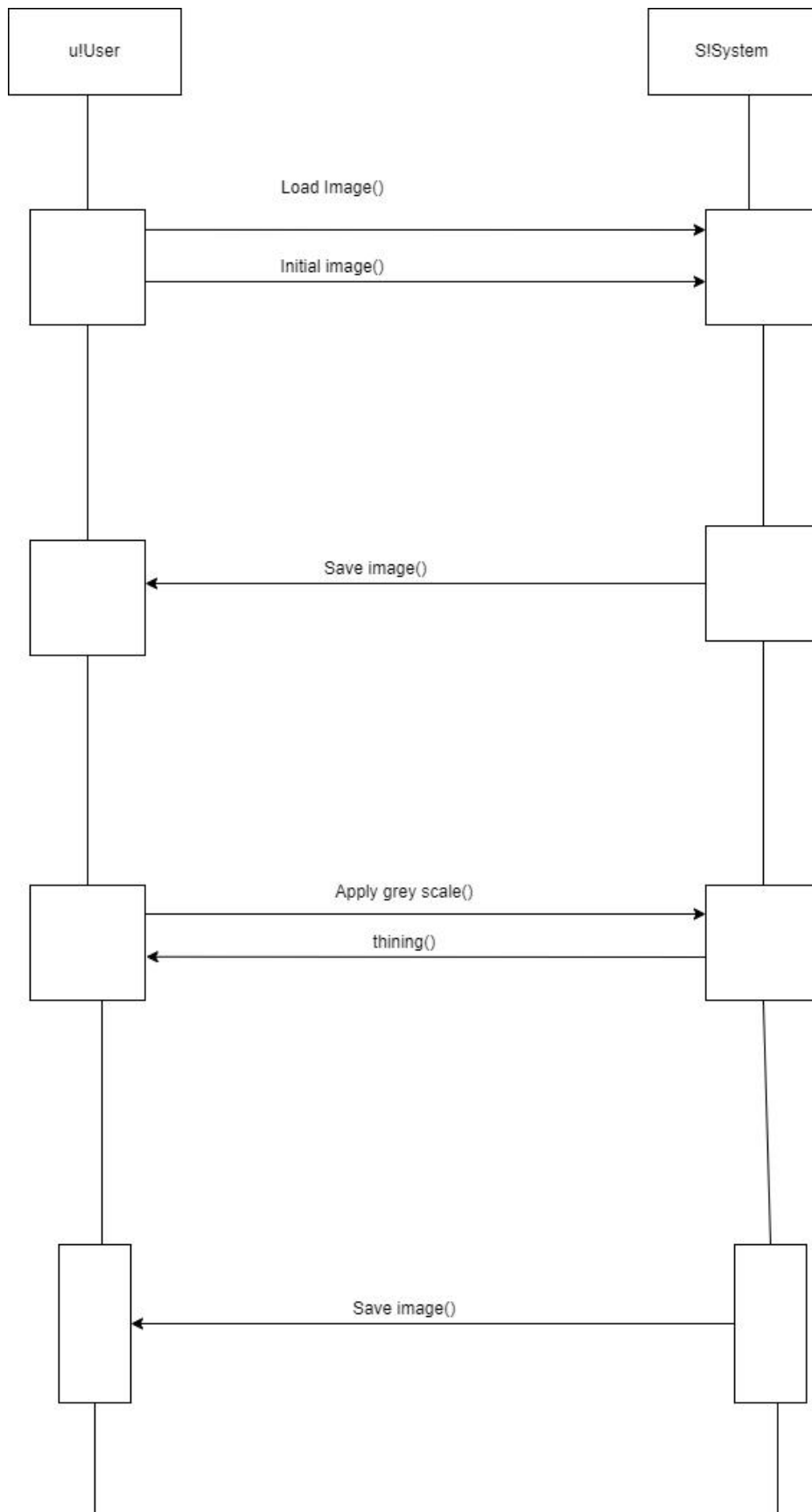
COMMUNICATION DIAGRAM
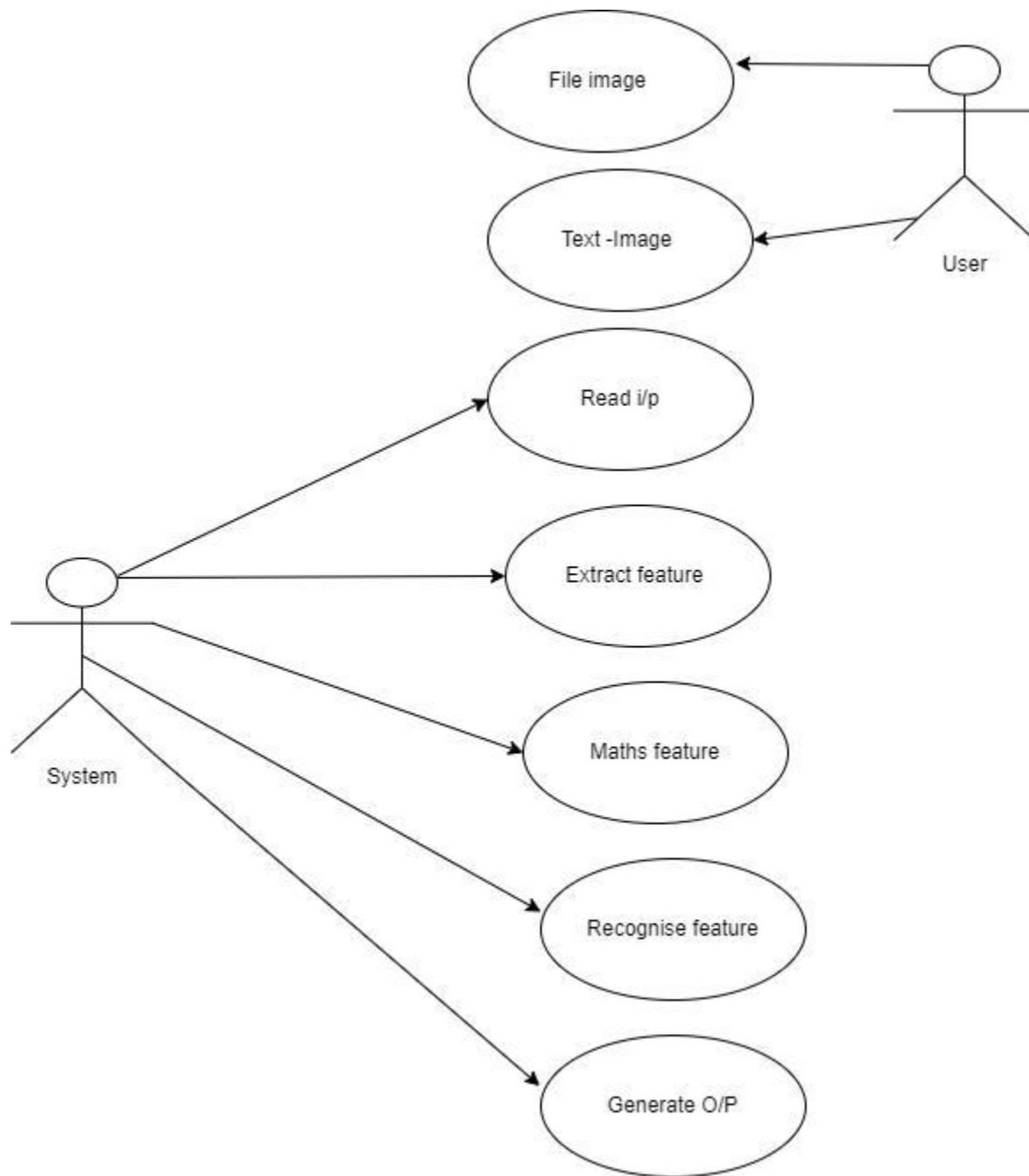
DEPOLYMENT DIAGRAM
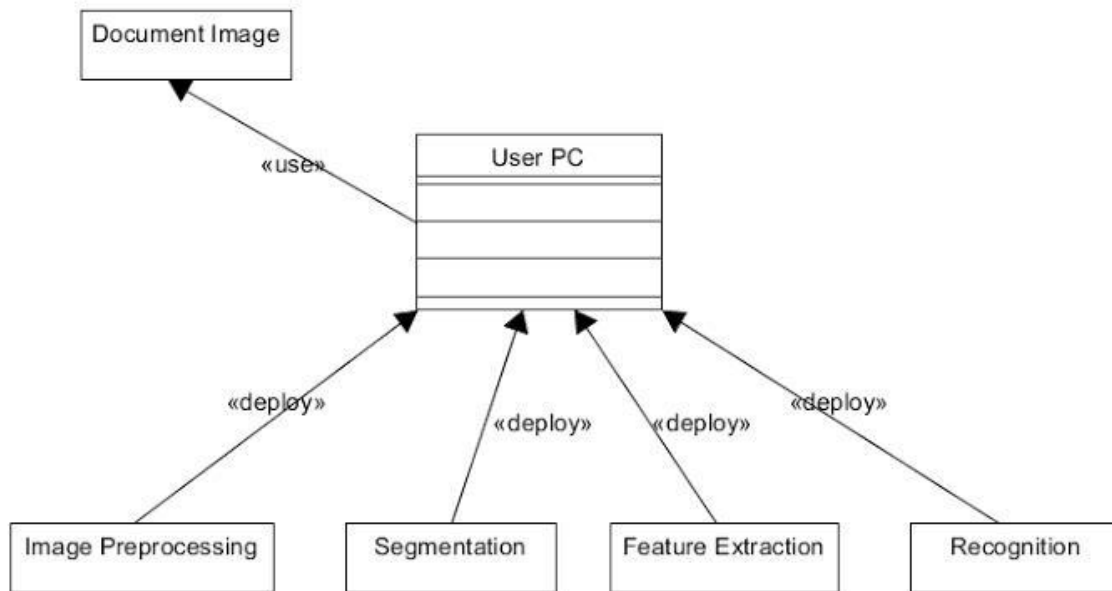
STATE MACHINE DIAGRAM
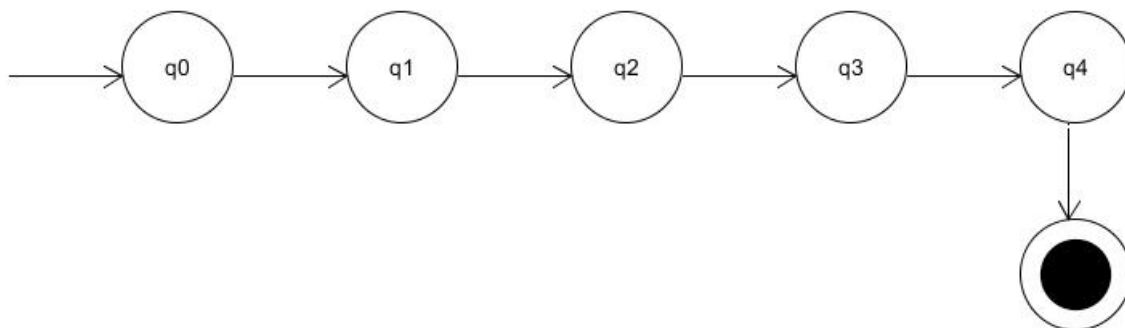
TIMING DIAGRAM
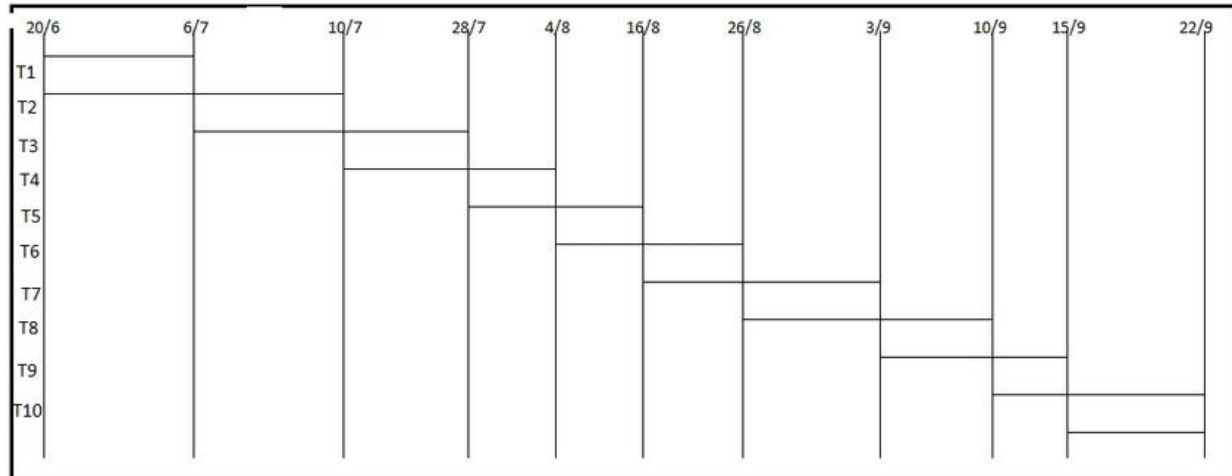
MODULE DESCRIPTION

MODULES

- User
- Can register to the website and then login.
- Can add a note that has to be converted to text message.
- Can view the converted file image and also hear the text message.

The proposed system can be achieved by implementing the following modules

- Image pre-processing
- Text localization
- Feature extraction
- Character recognition

IMAGE PRE-PROCESSING

Image pre-processing is carried out into steps such as colour conversion and Gaussian blurring. Color conversion function converts input image from one color space to other, here we used BGR2GRAY for converting the input image to gray scale image. The next step of pre-process is Gaussian blurring of images. Gaussian blurring removes noises from images and smoothest the images. For image segmentation, Adaptive Gaussian Threshold is applied and Threshold is calculated for every small region of images.

TEXT LOCALIZATION

Text localization can be done through image contours. Contours are used to analyze shape, object detection and recognition. Contours can be applied on binary images only, so image pre-process such as thresholding is performed before applying find Contours (). cv2.CHAIN_APPROX_SIMPLE () function is used to get the co-ordinate points for text localization.

FEATURE EXTRACTION

Using the above module, contours are identified, from which contour approximation is applied then bounding rectangle is identified as contour area. The segmented image is stored as temporary image, in which OpenCV function such as resize and reshape are applied to get yValue_images. The arrived yValue_images are considered as extracted feature.

- Adding 2D Convolutional layer

Add a 2D convolutional layer to process the 2D input images. The first argument passed to the Conv2D () layer function is the number of output channels – in this case we have 32 output channels. The next input is the kernel size, which in this case we have chosen to be a 5×5 moving window, followed by the strides in the x and y directions (1, 1). Next, the activation function is a rectified linear unit and finally we have to supply the model with the size of the input to the layer. Declaring the input shape is only required of the first layer – Kera's is good enough to work out the size of the tensors flowing through the model from there.

- Adding 2D max pooling layer

Add a 2D max pooling layer. We simply specify the size of the pooling in the x and y directions – (2, 2) in this case, and the strides.

- Adding another convolutional + max pooling layer

Next, we add another convolutional + max pooling layer, with 64 output channels. The default strides argument in the Conv2D () function is (1, 1) in Kera's, so we can leave it out. The default strides argument in Kera's is to make it equal to the pool size. The input tensor for this layer is (batch size, 28, 28, 32) – the 28 x 28 is the size of the image, and the 32 is the number of output channels from the previous layer.

- Flatten and adding dense layer

Next is to flatten the output from these to enter our fully connected layers. The next two lines declare our fully connected layers – using the Dense () layer in Kera's, we specify the size – in line with our architecture, we specify 1000 nodes, each activated by a ReLU function. The second is our soft-max classification, or output layer, which is the size of the number of our classes.

- Training neural network

In the training model, we have to specify the loss function, or told the framework what type of optimizer to use (i.e., gradient descent, Adam optimizer etc.).

Lass function of standard cross entropy for categorical class classification (karas. losses. categorical_crossentropy). We use the Adam optimizer (karas. optimizers. Adam). Finally, we can specify a metric that will be calculated when we run evaluate () on the model.

We first pass in all of our training data – in this case train and yttrian. The next argument is the batch size. In this case we are using a batch size of 32. Next, we pass the number of training epochs (2 in this case). The verbose flag, set to 1 here, specifies if you want detailed information being printed in the console about the progress of the training.

- Test data input

Finally, we pass the validation or test data to the fit function so Kera's knows what data to test the metric against when evaluate () is run on the model.

INPUT DESIGN

Input design is a part of the overall design. The input methods can be broadly classified. Internal controls must be established for monitoring the number of inputs and for ensuring that the data are valid. The basic steps involved in input design are:

1. Review input requirements.
2. Decide how the input data flow will be implemented.
3. Decide the source document.
4. Prototype online input screens.
5. Design the input screens.

    The quality of the system input determines the quality of the system output. Input specifications describe the way data enter the system for processing. Input design features can ensure the reliability of the system and produce result.

OUTPUT DESIGN

A quality is one, which meets the requirements of end user and present the information clearly. In any system results of processing are communicated to the user and to the other system through outputs. In the outputs design it is determined how the information is to be displayed for immediate need.

    It is the most important and direct source information is to user. Efficient and intelligent output design improves the system's relationships with the user and helps in decision making. The objectives of the output design are to convey the information of all the past activities,

current status and to emphasis important events. The output generally refers to the results and information that is generated from the system. Outputs from computes are required primarily to communicate the result of processing to the users.

   Output also provides a means of storage by coping the results for later reference in consolation. There is a chance that some of the end users will not actually operate the input data or information through workstations but will see the output from the system.

Two phases of the output design are:

1. Output Definitions
2. Output Specification

Output definitions considers the type of outputs contents, its frequency and its volume, the appropriate outputs media is determined for output. Once the media is chosen, the details specification of output documents are carried out. The nature of output required from the proposed system is determined during logical design stage. It takes the outline of the output from the logical design and produces output as specified during the logical design phase. In a project, when designing the output, the system must accomplish the following:

1. Determine the information to present.
2. Decide whether to display, speak the information and select the output medium.
3. Arrange the information in acceptable format.
4. Decide how to distribute the output to the intended receipt.

Thus, by following the above specification, a high-quality output can be generated. Outputs from compute system are required primarily to communicate thee result of processing to users. Computer output is the most important and direct source of information to the user. Efficiency, intelligible output should improve the system's relationship with the user and help in decision making. The output devices to consider depend on factors as compatibility of the device with the system, response time requirements, expected print quality, number of copies needed etc.

CHAPTER 4 SYSTEM ENVIRONMENT

INTRODUCTION

The system study is the process of gathering and interpreting facts, using this information for further studies on the system. It is a problem-solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies, a rough figure of the system activities can be obtained, from which the decisions about the strategies to be followed for effective system study and analysis can be taken. The system study also identifies the method collection to be followed.

The system study conducted an Initial picture about the system working was got. The information got form the study, the data collection methods are identified. Even in the first investigation itself drawbacks of the existing system could be identified.

SYSTEM SPECIFICATION

The software requirements specification (SRS) is a means of translating the ideas in the minds of clients into a formal documentation. This document forms the development and software validation. The basic reason for the difficulty in software requirement specification comes from the fact that there are three interested parties-the clients, the end users and the software developer. The requirements document has to be such that the client and the user can understand easily, and the developers can use it as a basis for software development. Due to the diverse parties involved in software requirement specification, a communication gap exists. This gap arises when the client does not understand software or the software development processor when the developer does not understand the client's problem and application area of SRS bridges this communication gap.

The software requirements specification (SRS) is a means of translating the ideas in the minds of clients into a formal documentation. This document forms the development and software validation. The basic reason for the difficulty in software requirement specification comes from the fact that there are three interested parties-the clients, the end users and the software developer. The requirements document has to be such that the client and the user can understand easily, and the developers can use it as a basis for software development. Due to the diverse parties involved in software requirement specification, a communication gap exists. This gap arises when the client does not understand software or the software development processor when the developer does not understand the client's problem and application area of SRS bridges this communication gap.

Problem analysis is done to obtain a clear understanding of the needs of the clients and the users, and what exactly is desired from the software. Analysis leads to the actual specification. People performing the analysis called analysts, area also responsible for specifying the requirements.

The software project is initiated by the client's needs. In the beginning these needs are in the minds of various people in the client organization. The requirement analyst must identify their requirements by talking to these people and understanding their needs. These people and the existing documents about the current mode of operation are the basis source of information for the analysts.

HARDWARE SPECIFICATION

Machine	Intel Dual Core
Mother Board	Intel 945 Chipset
Memory	4 GB
Hard Disk	500 GB
Monitor	18.5" LED Monitor
Keyboard	USB/3
Mouse	USB/3

SOFTWARE SPECIFICATION

Operating System	Windows
Web Technologies	Python, Django, HTML,CSS
Database	MySQL
Web Browser	Google Chrome/Mozilla Firefox

TOOLS AND PLATFORMS

FRONT END TOOLS

PYTHON

Python is a general-purpose interpreted, interactive, object-oriented, and highlevel programming language. It was created by Guido van Rossum during 1985-1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). This tutorial gives enough understanding on Python programming language. Python is a high-level, interpreted, interactive and object- oriented scripting language. Python is designed to

be highly readable. It uses English keywords frequently where as other languages use punctuation,and it has fewer syntactical constructions than other languages.

Python is a MUST for students and working professionals to become a great Software Engineer especially when they are working in Web Development Domain. I will list down some of the key advantages of learning Python. Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands. Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, Smalltalk, and UNIX shell and other scripting languages. Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL). Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library. Python was conceived in the late 1980s as a successor to the ABC language. Python 2.0, released in 2000, introduced features like list comprehensions and a garbage collection system capable of collecting reference cycles. Python 3.0, released in 2008, was a major revision of the language that is not completely backward-compatible, and much Python 2 code does not run unmodified on Python 3.The Python 2 language, i.e. Python 2.7.x, was officially discontinued on 1 January 2020 (first planned for 2015) after which security patches and other improvements will not be released for it. With Python 2's end- of-life, only Python 3.5.x and later are supported. Python interpreters are available for many operating systems. A global community of programmers develops and maintains Python, an open source reference implementation. A non-profit organization, the Python Software Foundation, manages and directs resources for Python and Python development.

Rather than having all of its functionality built into its core, Python was designed to be highly extensible. This compact modularity has made it particularly popular as a means of adding programmable interfaces to existing applications. Van Rossum's vision of a small core language with a large standard library and easily extensible interpreter stemmed from his frustrations with ABC, which espoused the opposite approach.

Python strives for a simpler, less-cluttered syntax and grammar while giving developers a choice in their coding methodology. In contrast to Perl's "there is more than one way to do it" motto, Python embraces a "there should be one and preferably only one obvious way to do it" design philosophy. Alex Martello, a Fellow at the Python Software foundation and Python book author, writer that "To describe something as 'clever' is not considered a compliment in the python culture." Python's developers strive to avoid premature optimization, and reject patches to non-critical part of the Python reference implementation that would offer marginal increases in speed at the cost of clarity. When speed is important, a Python programmer can move time-critical functions to extension modules written in languages such as C, or use Pippy, a just-in-time compiler. Python is also available, which translates a Python script into C and makes direct C-level API calls into the Python interpreter.

Why python?

Python has topped the charts in the recent years over other programming languages like C, C++ and Java and is widely used by the programmers. The language has undergone a drastic change since its release 25 years ago as many add-on features are introduced. The Python 1.0 had the module system of Modula-3 and interacted with Amoeba Operating System with varied functioning tools. Python 2.0 introduced in the year 2000 had features of garbage collector and Unicode Support. Python 3.0 introduced in the year 2008 had a constructive design that avoids duplicate modules and constructs. With the added features, now the companies are using Python 3.5The software development companies prefer Python language because of its versatile features and fewer programming codes. Nearly 14 % of the programmers use it on the operating systems like UNIX, Linux, Windows and Mac OS. The programmers of big companies use Python as it has created a mark for itself in the software development with characteristic features like-

• Interactive

• Interpreted

• Modular

• Dynamic

• Object-oriented

• Portable

• High level

• Extensible in C++ & C

Advantages of python:

Some of the benefits of programming in Python include:

• Presence of Third-Party Modules:

The Python Package Index (Pippi) contains numerous third-party modules that make Python capable

of interacting with most of the other languages and platforms.

• Extensive Support Libraries:

Python provides a large standard library which includes areas like internet protocols, string operations, web services tools and operating system interfaces.

• Open Source and Community Development:

Python language is developed under an OSI-approved open-source license, which makes it free to use and distribute, including for commercial purposes.

• User-friendly Data Structures:

Python has built-in list and dictionary data structures which can be used to construct fast runtime data structures.

• Learning Ease and Support Available:

Python offers excellent readability and uncluttered simple-to-learn syntax which helps beginners to utilize this programming language.

• Productivity and Speed:

Python has clean object-oriented design, provides enhanced process control capabilities, and possesses strong integration and text processing capabilities and its own unit testing framework, all of which contribute to the increase in its speed and productivity. Python is considered a viable option for building complex multi-protocol network applications.

WEB TECHNOLOGIES

Django

Django is a Python based free and open-source web framework, which follows the model-template-view (MTV) architectural pattern. Django's primary goal is to ease the creation of complex, database driven websites. The framework emphasizes reusability and pluggability of components, less code, low coupling, rapid development, and the principle of don't repeat yourself. Python is used throughout, even for settings files and data models. Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models.

Despite having its own nomenclature, such as naming the callable objects generating the HTTP responses views, the core Django framework can be seen as an MVC architecture. It consists of an object-relational mapper (ORM) that mediates between data models (defined as Python classes) and a relational database (Model), a system for processing HTTP requests with a web templating system (View), and a regular-expression-based URL dispatcher (Controller).

Also included in the core framework are:

- A lightweight and standalone web server for development and testing.
- A form serialization and validation system that can translate between HTML forms and values suitable for storage in the database.
- A template system that utilizes the concept of inheritance borrowed from object-oriented programming.
- A caching framework that can use any of several cache methods.
- Support for middleware classes that can intervene at various stages of request processing and carry out custom functions.
- An Internal dispatcher system that allows components of an application to communicate events to each other via pre-defined signal.
- An internationalization system, including translations of Django's own components into a variety of languages.
- A system for extending the capabilities of the template engine.
- An interface to Python's built-in unit test framework.
- Django REST framework is a powerful and flexible toolkit for building Web APIs.

<u>Hypertext Mark-up Language (HTML)</u>

It is the standard mark-up language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets.

Tags such as <imp /> and <input /> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags but use them to interpret the content of the page.

HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content.

<u>JavaScript</u>

It is a lightweight, interpreted, object-oriented language with first-class functions, and is best known as the scripting language for Web pages, but it's used in many non-browser

environments as well. It is a prototype-based, multi- paradigm scripting language that is dynamic, and supports object-oriented, imperative, and functional programming styles.

JavaScript runs on the client side of the web, which can be used to design / program how the web pages behave on the occurrence of an event. JavaScript is an easy to learn and powerful scripting language, widely used for controlling web page behaviour.

JavaScript can function as both a procedural and an object oriented language. Objects are created programmatically in JavaScript, by attaching methods and Properties to otherwise empty objects at run time, as opposed to the syntactic class definitions common in compiled languages like C++ and Java. Once an object has been constructed it can be used as a blueprint (or prototype) for creating similar objects

Cascading Style Sheets (CSS)

It is a style sheet language used to describe the presentation of a document written in HTML or XML (including XML dialects such as SVG, Math or XHTML). CSS describes how elements should be rendered on screen, on paper, in speech, or on other media.

BACK END TOOLS

MYSQL

MySQL is an Oracle-backed open source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually all platforms, including Linux, UNIX and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web applications and online publishing.

MySQL is an important component of an open-source enterprise stack called WAMP. WAMP is a web development platform that uses Linux as the operating system, Apache as the web server, and MySQL as the relational database management system and PHP as the object-oriented scripting language. (Sometimes Perl or Python is used instead of PHP.)

Originally conceived by the Swedish company MySQL AB, MySQL was acquired by Sun Microsystems in 2008 and then by Oracle when it bought Sun in 2010. Developers can use MySQL under the GNU General Public License (GPL), but enterprises must obtain a commercial license from Oracle. Today,MySQL is the RDBMS behind many of the top websites in the world and countless corporate and consumer-facing web-based applications, including Facebook, Twitter and YouTube.

MySQL is the database management system, or a database server.

How MySQL works

MySQL is based on a client-server model. The core of MySQL is MySQL server, which handles all of the database instructions (or commands). MySQL server is available as a

separate program for use in a client-server networked environment and as a library that can be embedded (or linked) into separate applications. MySQL operates along with several utility programs which support the administration of MySQL databases. Commands are sent to MySQL Server via the MySQL client, which is installed on a computer.

MySQL was originally developed to handle large databases quickly. Although MySQL is typically installed on only one machine, it is able to send the database to multiple locations, as users are able to access it via different MySQL client interfaces. These interfaces send SQL statements to the server and then display the results.

MySQL FEATURES

• Relational Database Management System (RDBMS): MySQL is a relational database management system

• Easy to use MySQL is easy to use: You have to get only the basic knowledge of SQL. You can build and interact with MySQL with only a few simple SQL statements.

• It is secure: MySQL consist of a solid data security layer that protects sensitive data from intruders. Passwords are encrypted in MySQL.

• Client/ Server Architecture: MySQL follows a client /server architecture. There is a database server (MySQL) and arbitrarily many clients (application programs), which communicate with the server; that is, they query data, save changes, etc.

• Free to download MySQL is free to use and you can download it from MySQL official website.

• It is scalable: MySQL can handle almost any amount of data, up to as much as 50 million rows or more. The default file size limit is about 4 GB. However, you can increase this number to a theoretical limit of 8 TB of data.

• Compatible on many operating systems: MySQL is compatible to run on many operating systems, like Novell NetWare, Windows* Linux*, many varieties of UNIX* (such as Sun* Solaris*, AIX, and DEC* UNIX), OS/2, FreeBSD*, and others. MySQL also provides a facility that the clients can run on the same computer as the server or on another computer (communication via a local network or the Internet).

• Allows roll-back: MySQL allows transactions to be rolled back, commit and crash recovery.

• High Performance: MySQL is faster, more reliable and cheaper because of its unique storage engine architecture.

• High Flexibility: MySQL supports a large number of embedded applications which makes MySQL very flexible.

• High Productivity: MySQL uses Triggers, Stored procedures and views which allows the developer to give a higher productivity.

Core MySQL features

MySQL enables data to be stored and accessed across multiple storage engines, including Inorb, CSV, and NDB. MySQL is also capable of replicating data and partitioning tables for better performance and durability. MySQL users aren't required to learn new commands; they can access their data using standard SQL commands. Before 2016, the main difference between MySQL and SQL was that the former could be used on multiple platforms, whereas the latter could only be used on Windows. Microsoft has since expanded SQL to support Linux, a change which went into effect in 2017. When MySQL is installed via Linux, its package management system requires custom configuration to adjust security and optimization settings.

MySQL also allows users to choose the most effective storage engine for any given table, as the program can utilize multiple storage engines for individual tables. One of MySQL's engines is Inorb. Inorb was designed for high availability. Because of this, it is not as quick as other engines. SQL uses its own storage system, but it does maintain multiple safeguards against loss of data. Both systems are able to run in clusters for high availability.SQL Server offers a wide variety of data analysis and reporting tools. SQL Server Reporting Services is the most popular one and is available as a free download. There are similar analysis tools for MySQL available from third-party software companies, such as Crystal Reports XI and Actuate BIRT.

PHPMYADMIN

PhpMyAdmin is a (web application) client for MySQL. MySQL is server where your commands get executed and returns you data, it manages all about data while PhpMyAdmin is a web Application, with user friendly, easy to use GUI makes it easy to handle database, which is difficult to use on command line. phpMyAdmin is the web application written primarily in PHP. It's used for managing MySQL database.

To be more specific, here is the detailed definition:

MySQL is the world's most popular open-source database. With its proven performance, reliability, and ease-of-use, MySQL has become the leading database choice for web-based applications, used by high profile web properties including Facebook, Twitter, YouTube, and all five of the top five websites. Additionally, it is an extremely popular choice as embedded database, distributed by thousands of ISVs and OEMs.

PhpMyAdmin is a free and open-source administration tool for MySQL and MariaDB. As a portable web application written primarily in PHP, it is one of the most popular MySQL administration tools, especially for web hosting services.

OPERATING SYSTEM

WINDOWS

Windows 10 professional integrates the strengths of windows 2008 professional such as standards-based security, manageability, and reliability, with the best business features of windows 98 and windows Millennium Edition, such as plug and play, simplified user interface, and innovative support services. This combination creates the best desktop operating system for business.It is more users friendly and a stable operating system equipped with much more added features. The operating system supports new technologies such as digital video disks, multiple monitors etc. along with plug and play and multi display features. It has a graphical user interface operating environment. Faster computing, easy access to remote information and control remote computers are some added features. Following are the common features of Windows 10.

Faster computing, easy access to remote information and control remote computers. Built-in networking and messaging facility.

- Easier to set up, add or remove.
- Increase system security and control.
- Support advanced networking and communication.

CHAPTER-5 SYSTEM IMPLEMENTATION

INTRODUCTION

A crucial phase in the system life cycle is the successful implementation of the new system design.

Implementation simply means converting a new system design into opera- action. This involves creating computer-compatible files, training the operating staff, and installing hardware, terminals and telecommunication network before the system is up and running.

In system implementation, user training is crucial for minimizing resistance to change and giving the new system a chance to prove its worth. Training aids, such as user-friendly manuals, a data dictionary, job performance aids that communicate information about the new system, and "HELP" screens, provide the user with a good start on the new system.

The term Implementation has different meanings, ranging from the conversion of a basic application to a complete replacement of a computer system. Implementation is used here to mean the process of converting a new or a revised system design into an operational one. Conversion is one aspect of implementation. Here, implementation of a modified application to replace an existing one, using the same computer is done.

The implementation stage involves following tasks:

• Careful planning

• Investigation of system and constraints.

• Design of method to achieve change over.

• Evaluation of the changeover method.

There are four methods,

• Parallel approach: The old system is operated with the new system.

• Direct cut over method: The old system is replaced with the new system.

• Pilot approach: Working version of the system is implemented in one part of the organization based on the feedback, changes are made, and the system is installed in the rest of the organization by one the other methods.

• Phase-in-method: Gradually implements the system across all users.

CODING

Coding tells a machine which actions to perform and how to complete tasks. Programming languages provide the rules for building websites, apps, and other computer-based technologies. Each programming language helps humans accurately communicate with machines.

Common styles are imperative, functional, logical, and object-oriented languages. Programmers can choose from these coding language paradigms to best-serve their needs for a specific project.

Coding Standards

Coding standards are important because they lead to greater consistency within code of all developers. Consistency leads to code that is easier to understand, which in turn results in a code, which is easier to develop and maintain. Code that difficult to understand and maintain runs the risk of being scrapped rewritten.

In my project, I follow the following coding standards,

• Use full English descriptors that accurately describe the variable/ field/ class/ interface.

 For example use variable like "username".

• Lengthy names are avoided.

• SQL commands starts with letters "cmd_".

 Eg:sqlCommand cmd;

• All connection variables starts with letters "con".

Sample code

HOME PAGE

{% extends 'user_header_footer.html' %}

```
{% block admin_home %}

{% load static %}


<!-- Top Navigation -->

{% if message %}

<script>

alert('{{message}}');

</script>

{% endif %}

<div class="banner">

        <div class="container">

        <script src="{% static 'js/responsiveslides.min.js' %}"></script>

 <script>

   $(function () {

    $("#slider").responsiveSlides({

        auto: true,

        nav: true,

        speed: 500,

      namespace: "callbacks",

      pager: true,

    });

   });

  </script>
```

```
<div class="slider">

            <div class="callbacks_container">

                <ul class="rslides" id="slider">

                    <li>

                        <h3>Nelson Mandela</h3>

                         <p>" Education is the most powerful weapon which you
can use to change the world "</p>

                    </li>

                    <li>

                        <h3>Benjamin Franklin</h3>

                         <p>" An investment in knowledge pays the best interest
"</p>

                    </li>

                    <li>

                        <h3>Clay P. Bedford</h3>

                         <p>" You can teach a student a lesson for a day;but if you
can teach him to learn by creating curiosity,he will continue the learning process as long as he
lives. "</p>

                    </li>

                    <li>

                        <h3>Dr. APJ Abdul Kalam</h3>

                         <p>" Some of the brightest minds in the country can be
found on the last benches of the classroom "</p>

                    </li>

                    <li>

                        <h3>Malala Yousafzai</h3>
```

```html
                                    <p>" One child,One teacher,One pen and One book can
change the world "</p>

                            </li>

                             <li>

                            <h3>Abraham Lincoln</h3>

                            <p>" The philosophy of the school room in one genaration
will be the philosophy of government in the next "</p>

                            </li>

                    </ul>

            </div>

        </div>

</div>

        </div>

<!--header-->

<!--weelcome-->

<div class="welcome">

        <div class="container">

                <form method="POST" enctype="multipart/form-data" >

                    {% csrf_token %}

                    <table>

                        <tr>

        <td>Text</td><td>

            <input type="text" name="t1">

        </td>
```

```
            </tr>

                              <tr>

                                    <td><input type="submit" name="b21"
value="Getword"></td>

                              </tr>


                  </table>

            </form>

      </div>

</div>

<!--/welcome-->

<div class="welcome-grids">

      <div class="container">

            <div class="welcome-gridsinfo">

            <div class="col-md-3 welcome-grid">

                  <i class="glyphicon glyphicon-pencil"> </i>

                  <h3>Prehistory</h3>

                  <p>Education began in the earliest prehistory, as adults trained the young
in the knowledge and skills deemed necessary in their.</p>

            </div>

            <div class="col-md-3 welcome-grid">

                  <i class="glyphicon glyphicon-font"> </i>

                  <h3>Prehistory</h3>

                  <p>Education began in the earliest prehistory, as adults trained the young
in the knowledge and skills deemed necessary in their.</p>
```

```html
                </div>

                <div class="col-md-3 welcome-grid">

                        <i class="glyphicon glyphicon-blackboard"> </i>

                        <h3>Prehistory</h3>

                        <p>Education began in the earliest prehistory, as adults trained the young
in the knowledge and skills deemed necessary in their.</p>

                </div>

                <div class="col-md-3 welcome-grid">

                        <i class="glyphicon glyphicon-education"> </i>

                        <h3>Prehistory</h3>

                        <p>Education began in the earliest prehistory, as adults trained the young
in the knowledge and skills deemed necessary in their.</p>

                </div>

                <div class="clearfix"> </div>

                </div>

        </div>

</div>

<!--effect-grid-->

<div class="effect-grid">

        <div class="container">


</div>

<!--\effect-grid-->

<!--testmonials-->
```

```
<div class="testimonials">

        <div class="container">

                    <div class="testimonial-nfo">

                            <h3>Testimonials</h3>

                            <h5>Cras porttitor imperdiet volutpat nulla malesuada lectus eros
<span>ut convallis felis consectetur ut </span></h5>

                    </div>

                    <div class="testimonial-grids">

                    <div class="testimonial-grid">

                            <p><span>"</span> Lorem ipsum dolor sit amet,
consectetur adipiscing elit. Nam fermentum iaculis diam quis sodales. Vestibulum eu dui tellus.
In viverra porttitor auctor. Pellentesque habitant morbi tristique senectus et netus et malesuada
fames ac turpis egestas<span> "</span></p>

                    </div>

                    </div>

        </div>

</div>

<!--\testmonials-->

<!--specfication-->

                    <div class="clearfix"> </div>

                    </div>

            </div>

        </div>

<!--/specfication-->

{% endblock %}
```

LOGIN PAGE

from Django. Shortcuts import render,HttpResponseRedirect,HttpResponse

from django.core.files.storage import FileSystemStorage

import webbrowser

import datetime

import matplotlib

matplotlib.use('Agg')

import matplotlib.pyplot as plt

import numpy as np

import random

import MySQLdb

db=MySQLdb.connect("localhost","root","","textdb")

c=db.cursor()

def index(request):

   return render(request,"index.html")

def login(request):

   msg=""

   if request.POST:

     uname=request.POST.get("email")

     password=request.POST.get("password")

     uname=uname.lower()

```
        password=password.lower()

        request.session['uname']=uname

        print(uname)

        print(password)

        query="select * from login where uname='"+str(uname)+"' and
password='"+str(password)+"'"

        c.execute(query)

        data=c.fetchone()

        print(data)

        if data:

            if data[2]=='admin':

                return HttpResponseRedirect("/adminhome/")

            elif data[2]=='user':

                if data[3]=="approved":

                    c.execute("select uid from userreg where email='"+str(request.session['uname'])+"'")

                    owner=c.fetchone()

                    request.session['uid']=owner[0]

                    return HttpResponseRedirect("/uhome/")

                else:

                    msg="WAIT FOR ADMIN APPROVAL.."

                    return render(request,"common/login.html",{"msg":msg})

        else:

            msg="invalid username or password"

    return render(request,"login.html")
```

```
def register(request):

    return render(request,"register.html")



def adminhome(request):

    return render(request,"adminhome.html")



def userhome(request):

    return render(request,"userhome.html")



def addnotes(request):

    uid=request.session['uid']

    if request.POST:

        description=request.POST.get("description")

        if request.FILES["file"]:

            myfile=request.FILES["file"]

            fs=FileSystemStorage()

            filename=fs.save(myfile.name,myfile)

            fileurl=fs.url(filename)

            query="insert into notes(notes,images,uid)
values('"+str(description)+"','"+str(fileurl)+"','"+str(uid)+"')"

            c.execute(query)

            db.commit()

    return render(request,"addnotes.html")
```

```python
def viewnotes(request):


    uid=request.session['uid']

    msg=""


    c.execute("select * from notes where uid='"+str(uid)+"'")

    data=c.fetchall()


    if request.GET:

        id=request.GET.get("id")

        c.execute("select * from notes where nid='"+str(id)+"'")

        data=c.fetchone()

        path="D:/PYTHON-PROJECTS-2022/textproject/textproject/textapp/static"+data[2]


        # import the following libraries
    # will convert the image to text string
        import pytesseract


        # adds image processing capabilities
        from PIL import Image


        # converts the text to speech
        import pyttsx3
```

#translates into the mentioned language

from googletrans import Translator

opening an image from the source path

img = Image.open(path)

describes image format in the output

path where the tesseract module is installed

pytesseract.pytesseract.tesseract_cmd = 'C:/Program Files/Tesseract-OCR/tesseract.exe'

converts the image to result and saves it into result variable

result = pytesseract.image_to_string(img)

print(result)

write text in a text file and save it to source path

with open('abc.txt',mode ='w') as file:

	file.write(result)

	print(result)

p = Translator()

translates the text into german language

k = p.translate(result,destde 'en')

print(k)

engine = pyttsx3.init()

```
        # an audio will be played which speaks the test if pyttsx3 recognizes it

        engine.say(k)

        engine.runAndWait()

        return HttpResponseRedirect("/viewnotes/")



    return render(request,"viewnote.html",{"data":data,"msg":msg})



def userreg(request):


    msg=""

    word=""

    if request.POST:

        name=request.POST.get("name")

        email=request.POST.get("email")

        address=request.POST.get("address")

        phoneno=request.POST.get("phoneno")

        password=request.POST.get("password")

        cpassword=request.POST.get("cpassword")

        if password==cpassword:

            status='approved'

            qq="select count(*) from userreg where email='"+str(email)+"'"

            c.execute(qq)
```

```
        data=c.fetchone()

        print(qq)

        print(data)



        if int(data[0])<1:

            query="insert into userreg(name,email,address,phoneno)
values('"+str(name)+"','"+str(email)+"','"+str(address)+"','"+str(phoneno)+"')"

            print(query)

            c.execute(query)

            db.commit()

            usertype='user'

            qqq="insert into login(uname,password,usertype,status)
values('"+str(email)+"','"+str(password)+"','"+str(usertype)+"','"+str(status)+"')"

            c.execute(qqq)

            db.commit()

            msg="Account successfully Created"

        else:

            msg="Allready have an account with same mail id"

    else:

        word="Sorry your password and confirm password are not matching"



    return render(request,"userreg.html",{"msg":msg,"word":word})
```

<u>REGISTRATION PAGE</u>

```
{% extends 'index_header_footer.html' %}

{% block registration %}

{% load static %}


{% if message %}

<script>

alert('{{message}}');

</script>

{% endif %}

<!-- Top Navigation -->

<div class="banner banner5">

        <div class="container">

        <h2>Registration</h2>

        </div>

</div>

<!--header-->

                <!-- contact -->

                <div class="contact">

                        <!-- container -->

                        <div class="container">


                                <div class="contact-grids">
```

<div class="contact-info cf-1">

<div class="contact-info-grids">

<div style="max-width: 150%; margin-left: 75px;">

<form method="POST" enctype="multipart/form-data">

{% csrf_token %}

<input type="text" name="name" pattern="[A-Za-z]{1}[A-Za-z\s]{2,32}" title="Please Enter characters only" placeholder="Name" required="" >

<input type="email" name="mail" placeholder="Email" required="" >

<input type="text" name="con" pattern="[A-Za-z\s]{1,32}" title="Please Enter characters only" placeholder="Country" required="">

<input type="text" name="addr" placeholder="Address" required="">

<input type="text" name="dis" pattern="[A-Za-z\s]{1,32}" title="Please Enter characters only" placeholder="District" required="">

<input type="text" name="phone" minlength="10" maxlength="10" pattern="[6789]{1}[0-9]{9}" placeholder="Contact No" required="">

<input type="password" name="password" maxlength="10" placeholder="password" required="" title="Enter password atleast 6 characters(first letter A-Z, second onwards a-z or 0-9)">

```
                                                        <input type="submit"
name="register" value="REGISTER">

                </form>

            </div>

                                            </div>

                            </div>

                    </div>

                </div>

                <!-- //container -->

            </div>

            <!-- //contact -->



            <!-- //container -->

        </div>

        <script>

    if ( window.history.replaceState ) {

        window.history.replaceState( null, null, window.location.href );

    }

  </script>

{% endblock %}
```

USER PAGE

```
<!doctype html>

<html>
```

```
<head>

<title>Object Recognize</title>

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<meta name="keywords" content="Education Tutorial Responsive web template, Bootstrap Web
Templates, Flat Web Templates, Andriod Compatible web template,

Smartphone Compatible web template, free webdesigns for Nokia, Samsung, LG,
SonyErricsson, Motorola web design" />

<script type="application/x-javascript"> addEventListener("load", function() {
setTimeout(hideURLbar, 0); }, false); function hideURLbar(){ window.scrollTo(0,1); }
</script>

<!--bootstrap-->

<link href="{% static 'css/bootstrap.css' %}" rel="stylesheet" type="text/css" media="all">

<!--coustom css-->




<link href="{% static 'css/style.css' %}" rel="stylesheet" type="text/css"/>

<!--script-->

<script src="{% static 'js/jquery-1.11.0.min.js' %}"></script>

<!-- js -->

<script src="{% static 'js/bootstrap.js' %}"></script>

<!-- /js -->

<!--fonts-->

<link
href='//fonts.googleapis.com/css?family=Open+Sans:300,300italic,400italic,400,600,600italic,70
0,700italic,800,800italic' rel='stylesheet' type='text/css'>
```

```
<!--/fonts-->

<!--hover-girds-->

<link rel="stylesheet" type="text/css" href="{% static 'css/default.css' %}" />

<link rel="stylesheet" type="text/css" href="{% static 'css/component.css' %}" />

<script src="js/modernizr.custom.js' %}"></script>

<!--/hover-grids-->

<script type="text/javascript" src="{% static 'js/move-top.js' %}"></script>

<script type="text/javascript" src="{% static 'js/easing.js' %}"></script>

<!--script-->

<script type="text/javascript">

                    jQuery(document).ready(function($) {

                            $(".scroll").click(function(event){

                                    event.preventDefault();


            $('html,body').animate({scrollTop:$(this.hash).offset().top},900);

                                    });

                            });

</script>

<!--/script-->

</head>

<body>

            <div class="header" id="home">

                            <nav class="navbar navbar-default">

                                    <div class="container">
```

<!-- Brand and toggle get grouped for better mobile display -->

<div class="navbar-header">

<button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-expanded="false">

Toggle navigation

</button>

<h1>OBJECT
Recognize system</h1>

</div>

<!-- Collect the nav links, forms, and other content for toggling -->

<div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">

<ul class="nav navbar-nav navbar-right margin-top cl-effect-2">

Home

<!--

View -->

{% comment %} Approve

`<li><a href="/view_userview">User Details</a></li> {% endcomment %}`

`<!-- <li><a href="contact.html"><span data-hover="Contact">Contact</span></a></li> -->`

`</ul>`

`<div class="clearfix"> </div>`

`</div><!-- /.navbar-collapse -->`

`<!-- /.container-fluid -->`

`<div class="login-pop">`

`<div id="loginpop"><a href="/" ><span>Logout</span></a>`

`</div>`

`</div><script src="{% static 'js/menu_jquery.js' %}"></script>`

`</div>`

`</nav>`

`<!--/script-->`

`<div class="clearfix"> </div>`

`</div>`

```
{% block admin_home %}

{% endblock %}


{% block approve_teacher %}

{% endblock %}


{% block view_teacher %}

{% endblock %}


{% block add_courses %}

{% endblock %}


{% block view_courses %}

{% endblock %}


{% block academic_year %}

{% endblock %}


{% block add_students %}

{% endblock %}


{% block view_students %}
```

```
{% endblock %}


   {% block add_subject %}

   {% endblock %}


   {% block add_exam %}

   {% endblock %}


   {% block a_exams %}

   {% endblock %}


   {% block a_view_attendance %}

   {% endblock %}


{% block t_view_students %}

    {% endblock %}



   {% block t_view_exam %}

    {% endblock %}
<!--footer-->
<br>

<br><br>
```

```html
<br><br>

<br>

<div class="footer">


        <!-- container -->

        <div class="container">

                <div class="col-md-6 footer-left">

                        <ul>

                                <li><a href="index.html">Ho</a></li>

                                <li><a href="about.html">About</a></li>

                                <li><a href="typography.html">Courses</a></li>

                                <li><a href="gallery.html">Gallery</a></li>

                                <li><a href="contact.html">Contact</a></li>

                        </ul>

                        <form>

                                <input type="text" placeholder="Email" required="">

                                <input type="submit" value="Subscribe">

                        </form>

                </div>

                <div class="col-md-3 footer-middle">

                        <h3>Address</h3>

                        <div class="address">

                                <p>Object
```

```
                              <span>G H Road,Ekm.</span>

                    </p>

            </div>

            <div class="phone">

                    <p>04865 236 273</p>

            </div>

        </div>

        <div class="col-md-3 footer-right">

                <h3>Object Info</h3>

                <p>Get Your Best recognition.</p>

        </div>

        <div class="clearfix"> </div>

    </div>

    <!-- //container -->

</div>

<!--/footer-->

<!--copy-rights-->

<div class="copyright">

    <!-- container -->

    <div class="container">

            <div class="copyright-left">

            <p>© 2020 Object Recognize. All rights reserved | Design by <a
href="http://w3layouts.com/">PRAVEEN</a></p>

            </div>
```

```html
<div class="copyright-right">

    <ul>

        <li><a href="#" class="twitter"> </a></li>

        <li><a href="#" class="twitter facebook"> </a></li>

        <li><a href="#" class="twitter chrome"> </a></li>

        <li><a href="#" class="twitter pinterest"> </a></li>

        <li><a href="#" class="twitter linkedin"> </a></li>

        <li><a href="#" class="twitter dribbble"> </a></li>

    </ul>

</div>

<div class="clearfix"> </div>


</div>

<!-- //container -->

<!---->
<script type="text/javascript">

    $(document).ready(function() {

        /*

        var defaults = {

        containerID: 'toTop', // fading element id

        containerHoverID: 'toTopHover', // fading element hover id

        scrollSpeed: 1200,

        easingType: 'linear'
```

```
                                };

                                */

                $().UItoTop({ easingType: 'easeOutQuart' });

});

</script>

<a href="#to-top" id="toTop" style="display: block;"> <span id="toTopHover" style="opacity:
1;"> </span></a>

<!---->

        </div>

<!--/copy-rights-->

        </body>

</html>
```

VIEW PAGE

```
<!doctype html>

<html>

<head>

<title>Object Recognize</title>

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<meta name="keywords" content="Education Tutorial Responsive web template, Bootstrap Web
Templates, Flat Web Templates, Andriod Compatible web template,

Smartphone Compatible web template, free webdesigns for Nokia, Samsung, LG,
SonyErricsson, Motorola web design" />

<script type="application/x-javascript"> addEventListener("load", function() {
setTimeout(hideURLbar, 0); }, false); function hideURLbar(){ window.scrollTo(0,1); }
</script>
```

```
<!--bootstrap-->

<link href="{% static 'css/bootstrap.css' %}" rel="stylesheet" type="text/css" media="all">

<!--coustom css-->



<link href="{% static 'css/style.css' %}" rel="stylesheet" type="text/css"/>

<!--script-->

<script src="{% static 'js/jquery-1.11.0.min.js' %}"></script>

<!-- js -->

<script src="{% static 'js/bootstrap.js' %}"></script>

<!-- /js -->

<!--fonts-->

<link href='//fonts.googleapis.com/css?family=Open+Sans:300,300italic,400italic,400,600,600italic,700,700italic,800,800italic' rel='stylesheet' type='text/css'>

<!--/fonts-->

<!--hover-girds-->

<link rel="stylesheet" type="text/css" href="{% static 'css/default.css' %}" />

<link rel="stylesheet" type="text/css" href="{% static 'css/component.css' %}" />

<script src="js/modernizr.custom.js' %}"></script>

<!--/hover-grids-->

<script type="text/javascript" src="{% static 'js/move-top.js' %}"></script>

<script type="text/javascript" src="{% static 'js/easing.js' %}"></script>

<!--script-->
```

```
<script type="text/javascript">

                        jQuery(document).ready(function($) {

                                $(".scroll").click(function(event){

                                        event.preventDefault();


        $('html,body').animate({scrollTop:$(this.hash).offset().top},900);

                                });

                        });

</script>

<!--/script-->

</head>

<body>

                <div class="header" id="home">

                                <nav class="navbar navbar-default">

                                        <div class="container">

                                                <!-- Brand and toggle get grouped for better mobile
display -->

                                                <div class="navbar-header">

                                                <button type="button" class="navbar-toggle
collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-
expanded="false">

                                                        <span class="sr-only">Toggle
navigation</span>

                                                        <span class="icon-bar"> </span>

                                                        <span class="icon-bar"> </span>

                                                        <span class="icon-bar"> </span>
```

```
                                                    </button>

                                                    <h1><a class="navbar-brand"
href="/admin_home">OBJECT<br /><span>Recognize system</span></a></h1>

                                                    </div>

                                                    <!-- Collect the nav links, forms, and other content
for toggling -->

                                                    <div class="collapse navbar-collapse"
id="bs-example-navbar-collapse-1">

                                                    <ul class="nav navbar-nav navbar-
right margin-top cl-effect-2">

                                                    <li><a
href="/adminhome"><span data-hover="Home">Home</span></a></li>

                                                    <!-- <li><a href="/recog">View</a></li> --
>

                                                    <li><a
href="/approve_teacherview">Approve</a></li>

                                                    <li><a
href="/view_userview">User Details</a></li>

                                                    <!-- <li><a
href="contact.html"><span data-hover="Contact">Contact</span></a></li> -->

                                                    </ul>

                                                    <div class="clearfix"> </div>
```

```
                                        </div><!-- /.navbar-collapse -->

                        <!-- /.container-fluid -->

                        <div class="login-pop">

                                        <div id="loginpop"><a href="/"
><span>Logout</span></a>

                                        </div>

                                        </div><script src="{% static
'js/menu_jquery.js' %}"></script>

                                        </div>

                        </nav>

        <!--/script-->

                        <div class="clearfix"> </div>

        </div>


  {% block admin_home %}

  {% endblock %}


  {% block approve_teacher %}

  {% endblock %}


  {% block view_teacher %}

  {% endblock %}


  {% block add_courses %}
```

```
{% endblock %}


{% block view_courses %}

{% endblock %}


{% block academic_year %}

{% endblock %}


{% block add_students %}

{% endblock %}


{% block view_students %}

{% endblock %}


{% block add_subject %}

{% endblock %}


{% block add_exam %}

{% endblock %}


{% block a_exams %}

{% endblock %}
```

```
{% block a_view_attendance %}

{% endblock %}


{% block t_view_students %}

    {% endblock %}



    {% block t_view_exam %}

    {% endblock %}

<!--footer-->

<br>

<br><br>

<br><br>

<br>

<div class="footer">


        <!-- container -->

        <div class="container">

                <div class="col-md-6 footer-left">

                        <ul>

                                <li><a href="index.html">Ho</a></li>

                                <li><a href="about.html">About</a></li>

                                <li><a href="typography.html">Courses</a></li>
```

```html
            <li><a href="gallery.html">Gallery</a></li>

            <li><a href="contact.html">Contact</a></li>

        </ul>

        <form>

            <input type="text" placeholder="Email" required="">

            <input type="submit" value="Subscribe">

        </form>

    </div>

    <div class="col-md-3 footer-middle">

        <h3>Address</h3>

        <div class="address">

            <p>Object

                <span>G H Road,Ekm.</span>

            </p>

        </div>

        <div class="phone">

            <p>04865 236 273</p>

        </div>

    </div>

    <div class="col-md-3 footer-right">

        <h3>Object Info</h3>

        <p>Get Your Best recognition.</p>

    </div>
```

```
<div class="clearfix"> </div>

</div>

<!-- //container -->

</div>

<!--/footer-->

<!--copy-rights-->

<div class="copyright">

<!-- container -->

<div class="container">

<div class="copyright-left">

<p>© 2020 Object Recognize. All rights reserved | Design by <a
href="http://w3layouts.com/">PRAVEEN</a></p>

</div>

<div class="copyright-right">

<ul>

<li><a href="#" class="twitter"> </a></li>

<li><a href="#" class="twitter facebook"> </a></li>

<li><a href="#" class="twitter chrome"> </a></li>

<li><a href="#" class="twitter pinterest"> </a></li>

<li><a href="#" class="twitter linkedin"> </a></li>

<li><a href="#" class="twitter dribbble"> </a></li>

</ul>

</div>

<div class="clearfix"> </div>
```

```
</div>

<!-- //container -->

    <!---->
<script type="text/javascript">

        $(document).ready(function() {

            /*

            var defaults = {

            containerID: 'toTop', // fading element id

            containerHoverID: 'toTopHover', // fading element hover id

            scrollSpeed: 1200,

            easingType: 'linear'

            };

            */

        $().UItoTop({ easingType: 'easeOutQuart' });

});

</script>

<a href="#to-top" id="toTop" style="display: block;"> <span id="toTopHover" style="opacity: 1;"> </span></a>

<!---->

    </div>

<!--/copy-rights-->

    </body>

</html>
```

Code Validation and Optimization

A validation check ascertains that the value (or data) input into a computer is valid. Validation checks are performed automatically by computer to ensure that entered data is correct and reasonable. Validation means check the input submitted by the user. There are two types of validation are available in PHP. They are as follows –

• Client-Side Validation – Validation is performed on the client machine web browsers.

• Server-Side Validation – After submitted by data, the data has sent to a server and perform validation checks in server machine.

• Server-Side Validation

In the Server-Side Validation, the input submitted by the user is being sent to the server and validated using one of server-side scripting languages such as ASP.Net, PHP etc. After the validation process on the Server Side, the feedback is sent back to the client by a new dynamically generated web page. It is better to validate user input on Server Side because you can protect against the malicious users, who can easily bypass your Client-Side scripting language and submit dangerous input to the server.

• Client-Side Validation

In the Client-Side Validation, you can provide a better user experience by responding quickly at the browser level. When you perform a Client-Side Validation, all the user inputs validated in the user's browser itself. Client-Side validation does not require a round trip to the server, so the network traffic which will help your server perform better. This type of validation is done on the browser side using script languages such as JavaScript, VBScript or HTML5 attributes.

Debugging

Debugging is the process of finding and fixing errors or bugs in the source code of any software. When software does not work as expected, computer programmers study the code to determine why any errors occurred.

Debugging steps:

i. Reproduce the problem.

ii.Describe the bug. Try to get as much input from the user to get the exact reason.

iii.Capture the program snapshot when the bug appears. ...

iv.Analyse the snapshot based on the state and action. ...

v.Fix the existing bug, but also check that any new bug does not occur.

Debugging techniques:

There are two types of debugging techniques:

• Reactive debugging

• Preemptive debugging.

• **Reactive Debugging**

Reactive debugging refers to any debugging protocol that is employed after the bug manifests itself. Reactive debugging is deployed to reduce runtime and logic errors. Examples of reactive debugging are print debugging and using a debugger.

• **Preemptive Debugging**

Preemptive debugging involves writing code that doesn't impact the functionality of the program but helps developers, either catch bugs sooner or debug the source code easily when the bug occurs.

Importance of debugging:

Debugging is important because it allows software engineers and developers to fix errors in a program before releasing it to the public. It's a complementary process to testing, which involves learning how an error affects a program overall.

Unit testing

Unit testing is a level of a software testing where individual units/components of software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing.

Test Plan and Test Cases

Test plan:

Test Plan is a detailed document that catalogs the test strategies, objectives, schedule, estimations, deadlines, and resources required to complete that project. Think of it as a blueprint for running the tests needed to ensure the software is working correctly – controlled by test managers. Writing a test plan is typically a test management or leadership responsibility. Others on the test team and in the organization (such as users and developers) may have input and review tasks, but it is generally up to the manager to actually write the test plan.

Types of Test Plans

i. Master Test Plan

The master test plan is a document that goes into great depth on the planning and management of testing at various test levels. It provides a bird's eye view of the important choices made, the tactics used, and the testing effort put forth for the project. The master test plan includes the list of tests that must be run. Test coverage, connections between various test levels and associated code tasks, test execution strategies, etc.

ii.Test Phase Plan

The testing plans that must be followed for each test level, or occasionally test type, are described in detail in the level test plan. The level test plan typically includes further information on the levels listed in the master testing plan. They would offer the testing schedule, benchmarks, activities, templates, and other information that isn't included in the master plan.

iii. Test Plans

Plans for conducting particular testing, such as performance and security tests. For instance, performance testing is software testing that aims to ascertain how a system responds and performs under a specific load. Security testing is software testing that aims to ascertain the system's vulnerabilities and whether its data and resources are safe from potential intruders.

Objectives of Software Test Plan:

• It provides an overview of where to start and stop the work.

• To be precise about the number of resources needed to finish the work.

• Timeline, based on the number of hours and workers needed.

• It has every detail from beginning to end, just like a prototype.

• To create the detailed tasks that must be carried out in the project's modules.

• It serves as a guide for following rules when the project is completed phase by phase.

• It will assist in considering the project's challenges and identifying solutions.

• All stakeholder interactions will be taken into consideration and organized.

Steps to write an Effective Test Plan

These sections should be covered for an effective test plan:

Step 1: Test Plan Identifier

The name of the document, its version number, and the year it was created are all included in the title of the test plan document, along with the name and corporate logo of the QA provider.

Step 2: Introduction

Here, we give a quick description of our project plan. A note to a client serves as the introduction.

Step 3: Test Items

The overall functionality to be tested is a test item and includes installation, registration, checkout, etc. It serves as a concise summary of the Test Plan's information. Each of the items will be covered in more detail later. Depending on the objective or type of testing, the list may need to be cut or increased.

Step 4: Features to be Tested

This section contains a more detailed list of features to inspect throughout a given period, which is what most people think a test plan should have.

Step 5: Approach

As a result, a client can obtain a thorough understanding of the testing activities. We outline the methods and types of testing we'll employ. This section also contains test cases.

Step 6: Test Deliverables

We inform a client about the materials they will receive to see the work's results, as the term implies. Test results are typically presented in test deliverables as metrics, such as the number of tests run, defects discovered, etc. Metrics are, in a sense, quantitative measures of quality, but they shouldn't be the only standard used to gauge the quality of the work.

Step 7: Staffing and Training Needs

The team might not fully comprehend the details if we are given the task of testing software for nuclear reactors. It is useful to have a lecture or brief course from experts when the team is testing a project from a sector they are unfamiliar with, putting exaggerations aside. Understanding the specifics of a project will help increase the task's effectiveness.

Step 8: Schedule

Include project deadlines in the project plan. The team must assess the speed or the time it will take to complete the tests. If there are many testing phases, specify their time and order.

Step 9: Planning Risks and Contingencies

The Software Risk Issues section and this part crossover. We list the risks and explain how to manage them and what to do in case of force majeure, in addition to listing the dangers.

Test cases

A test case refers to the actions required to verify a specific feature or functionality in software testing. The test case details the steps, data, prerequisites, and post conditions necessary to verify a feature.

Types of Test Cases:

i. Functionality Test Case: The functionality test case is to determine if the interface of the software works smoothly with the rest of the system and its users or not. Black box testing is used while checking for this test case, as we check everything externally and not internally for this test case.

ii. Unit Test Case: In unit test case is where the individual part or a single unit of the software is tested. Here each unit/ individual part is tested and we create a different test case for each unit.

iii.User Interface Test Case: The UI test or user interface test is when every component of the UI that the user would come in contact with is tested. It is to test if the UI components requirement made by the user are fulfilled or not.

iv.Integration Test Case: Integration testing is when all the units of the software are combined together and then they are tested. It is to check that each component and its units work together without any issues.

v.Performance Test Case: The performance test case helps to determine response time as well as the overall effectiveness of the system/software. It's to see if the application will actually handle the real-world expectations.

vi. Database Test Case: Also known as back-end testing or data testing checks that everything works fine with respect to the database. Testing cases for tables, schema, triggers, etc. Are done.

vii.Security Test Case: The security test case helps to determine that the application restricts actions as well as permissions wherever necessary. Encryption and authentication are considered as main objectives with regard to the security test case. The security test case is done to protect and safeguard the data of the software.

viii.Usability Test Case: Also known as a user experience test case, it checks how user-friendly or easy to approach a software would be. Usability test cases are designed by the User experience team and performed by the testing team.

ix.User Acceptance Test Case: The user acceptance case are prepared by the testing team but the user/client does the testing and review if they work in the real world environment.

Advantages of test case

There are many advantages of writing test cases:

• Test cases help to check if a particular module/software is meeting the specified requirement or not.

• Test cases determine if a particular module/software work with a given set of conditions.

• Test cases help to narrow down the software needs and required updates.

• Test cases are easy, simple, and clear as they are step by step and well documented.

• Test cases are detailed which makes them helpful during the maintenance phase.

CHAPTER 6 SYSTEM TESTING

INTRODUCTION

The term software testing is defined as to find for the errors in the application that might lead to fault or failure of the whole application. There are testing conditions that the system must pass to says that it is tested and working properly. The quality and reliability are also attained by going through the process of testing.

Integration Testing

Integration testing is a level of software testing where individual units are combined and tested as a group. The purpose of the level of testing is to expose faults in the interaction between integrated units. The purpose of this level of testing is to expose faults in the intersection between integrated units. The drivers and test stubs are used to assist in Integration Testing.

System Testing

System testing is a level of software testing where complete and integrated software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements.

Unit Testing

Unit testing is a level of a software testing where individual units/components of software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing.

Acceptance Testing

Acceptance testing is performed to ensure that the functional, behavioural, and performance requirements of the software are met IEEE defines acceptance testing as a 'formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system.'

During acceptance testing, the software is tested and evaluated by a group of users either at the developer's site or user's site. This enables the users to site. The enables the users to test the software themselves and analyses whether it is meeting their requirements. To perform acceptance testing, a predetermined set of data is given to the software as input. It is important to know the expected output before performing acceptance testing so that outputs produced by the software as a result of testing can be compared with them. Based on the results of tests, users decide whether to accept or reject the software is correct and is accepted; otherwise, it is rejected.

Regression Testing

Regression testing is the retesting of a software system to confirm that changes made to few parts of the codes has not any side effects on existing system functionalities. It is to ensure that old codes are still working as they were before introduction of the new change. The ideal process would be to create an extensive test suite and run it after each and every change.

CHAPTER 7 SYSTEM MAINTENANCE

INTRODUCTION

Maintenance means restoring something to its original conditions. Enhancement means adding, modifying the code to support the changes in the user specification. System maintenance conforms the system to its original requirements and enhancement adds to system capability by incorporating new requirements.

Thus, maintenance changes the existing system, enhancement adds features to the existing system, and development replaces the existing system. It is an important part of system development that includes the activities which corrects errors in system design and implementation, updates the documents, and tests the data.

Maintenance Types

System maintenance can be classified into four types –

i. Corrective Maintenance

ii. Adaptive Maintenance

iii. Perfective Maintenance

iv. Preventive Maintenance

i. Corrective Maintenance

Corrective Maintenance deals with the repair of faults or defects found in day- today system functions. A defect can result due to errors in software design, logic and coding. Design errors occur when changes made to the software are incorrect, incomplete, wrongly communicated, or the change request is misunderstood. Logical errors result from invalid tests and conclusions, incorrect implementation of design specifications, faulty logic flow, or incomplete

implementation of design specifications, faulty logic flow, or incomplete test of data. All these errors, referred to as residual errors, prevent the software from confirming to its agreed specifications. Note that the need for corrective maintenance is usually initiated by big reports drawn by the users.

ii. Adaptive Maintenance

Adaptive Maintenance is the implementation of changes in a part of the system, which has been affected by a change that occurred in some other part of the system. Adaptive Maintenance consists of adapting software to changes in the environment such as the hardware or the operating system. The term environment in this context refers to the conditions and the influences which act (from outside) on the system. For example, business rules, work patterns and government policies have a significant impact on the software system.

iii. Perfective Maintenance

Perfective Maintenance mainly deals with implementing new or changed user requirements. Perfective Maintenance involves making functional enhancements to the system in addition to the activities to increase the system's performance even when the changes have not been suggested by faults. This includes enhancing both the function and efficiency of the code and changing the functionalities of the system as per the users' changing needs.

iv. Preventive Maintenance

Preventive Maintenance involves performing activities to prevent the occurrence of errors. It tends to reduce the software complexity thereby improving program understand ability and increasing software maintainability. It comprises documentation updating, code optimization and code restructuring. Documentation updating involves modifying the documents affected by the changes in order to correspond to the present state of the system. Code optimization involves modifying the programs for faster execution or efficient use of storage space. Code restructuring involves transforming the program structure for reducing the complexity in source code and making it easier to understand.

CHAPTER 8 SYSTEM SECURITY MEASURES

INTRODUCTION

The protection of computer-based resources that includes hardware, software, data, procedures and against unauthorized use or natural.

• Integrity: Integrity refers to the power functioning of hardware and programs, appropriate physical security, and safety against external threats such as eavesdropping and writes tapping.

• Privacy

• Disaster is known as system security.

• System security can be divided into four related issues.

• Confidentiality

• Security

   Data security is the protection of data from loss, disclosure, modification and destruction.

System Integrity refers to the power functioning of hardware and programs, appropriate physical security, and safety against external threats such as eavesdropping and writes tapping.

Operating System Level Security

Operating system security (OS security) is the process of ensuring OS integrity, confidentiality and availability. OS security refers to specified steps or measures used to protect the OS from threats, viruses, worms, malware or remote hacker intrusions.

Database Level Security

Database security refers to the range of tools, controls, and measures designed to establish and preserve database confidentiality, integrity and availability. This article will focus primarily on confidentiality since it's the element that's compromised in most data breaches. There are three layers of database security: the database level, the access level, and the perimeter level. Security at the database level occurs within the database itself, where the data live. Access layer security focuses on controlling who can access certain data or systems containing it. Security policy at the perimeter level determines who can and cannot get into databases. Each level requires unique security solutions.

System Level Security

System-level security controls include the ability to assign administrative roles, assign application user privileges, and specify credentials that enable the system components to access data sources. There are three primary areas or classifications of system security controls. These include management security, operational security, and physical security controls.

CHAPTER 9 SYSTEM PLANNING AND SCHEDULING

INTRODUCTION

System planning and scheduling of your resource plays a key role in project management. It helps you understand the scope of the project ahead of time and manage/assign your resource accordingly. Besides, it provides an overview of who's responsible for delivering what and by when. In other words, you get a clear picture of what's happening through the lifecycle of a project and manage each task closely through collaboration and clear communication.

Planning a Software Project

Planning begins with the initiation of a project when you define its scope and purpose. You must already have your project team in place by this point along with an outline business case and a detailed understanding of your client/stakeholders' expectations. Planning your resources also involves analyzing the risk log.

At this point, you need to broadly plan the project and determine the cost associated with it while defining the "who, what, why, when and how" of the particular project. As development begins, you walk into a more thorough stage of planning where you outline clear specifications such as what you need to do, who will be responsible and by when. This helps you to sensibly manage your resources, time and budget and prepare a 'schedule' based on that.

Components:

The three major parts of a project plan are the scope, budget and timeline. They involve the following aspects:

• Scope. The scope determines what a project team will and will not do. It takes the team's vision, what stakeholders want and the customer's requirements and then determines what's possible. As part of defining the project scope, the project manager must set performance goals.

• Budget. Project managers look at what manpower and other resources will be required to meet the project goals to estimate the project's cost.

• Timeline. This reveals the length of time expected to complete each phase of the project and includes a schedule of milestones that will be met.

Project Lifecycle:

Projects typically pass through five phases. The project lifecycle includes the following:

• Initiation defines project goals and objectives. It also is when feasibility is considered, along with how to measure project objectives.

• Planning sets out the project scope. It establishes what tasks need to get done and who will do them.

• Execution is when the deliverables are created. This is the longest phase of a project. During execution, the plan is set into motion and augmented, if necessary.

• Monitoring and management occur during the execution phase and may be considered part of the same step. This phase ensures that the project is going according to plan.

• Closing and review is the final Contracts are closed out and the final deliverables are given to the client. Successes and failures are evaluated.

Steps Involved in Planning a System

Project planning includes the following 10 steps:

i.Define stakeholders. Stakeholders include anyone with an interest in the project. They can include the customer or end user, members of the project team, other people in the organization the project will affect and outside organizations or individuals with an interest.

ii. Define roles. Each stakeholder's role should be clearly defined. Some people will fill multiple roles, however.

iii. Introduce stakeholders. Hold a meeting to bring stakeholders together and unify the vision behind the project. The topics covered should include scope, goals, budget, schedule and roles.

iv. Set goals. Take what is gleaned from the meeting and refine it into a project plan. It should include goals and deliverables that define what the product or service will result in.

v. Prioritize tasks. List tasks necessary to meet goals and prioritize them based on importance and interdependencies. A Gantt chart can be helpful for mapping project dependencies.

vi. Create a schedule. Establish a timeline that considers the resources needed for all the tasks.

vii. Assess risks. Identify project risks and develop strategies for mitigating them.

viii. Communicate. Share the plan with all stakeholders and provide communications updates in the format and frequency stakeholders expect.

ix. Reassess. As milestones are met, revisit the project plan and revise any areas that are not meeting expectations.

x. Final evaluation. Once the project is completed, performance should be evaluated to learn from the experience and identify areas to improve.

Gantt chart

A Gantt chart is a project management tool that illustrates work completed over a period of time in relation to the time planned for the work. It typically includes two sections: the left side outlines a list of tasks, while the right side has a timeline with schedule bars that visualize work. The Gantt chart can also include the start and end dates of tasks, milestones, dependencies between tasks, and assignees.

Gantt Chart

Task Name	Q1 2019			Q2 2019		Q3 2019
	Jan 19	Feb 19	Mar 19	Apr 19	Jun 19	Jul 19
Planning	�					
Research		▉bar				
Design			▉bar			
Implementation				▉bar		
Follow up						▉bar

Pert chart

A PERT chart, sometimes called a PERT diagram, is a project management tool used to schedule, organize and coordinate tasks within a project. It provides a graphical representation of a project's timeline that enables project managers to break down each individual task in the project for analysis.

CHAPTER 10 SYSTEM COST ESTIMATION

INTRODUCTION

The aim of effective project management is to bring the project time completion on time and on schedule. Estimating project duration is a key function of scheduling. Individual

LOC BASED ESTIMATION

The LOC is a product size metric in software engineering .Here ,the number of lines in the code are counted and based on the number of lines the cost is calculated.

CHAPTER 11 FUTURE ENHANCEMENT AND SCOPE OF FURTHER DEVELOPMENT

INTRODUCTION

Making enhancements is all about perfective maintenance. It means adding, modifying or redeveloping the code to support changes in the specifications. It is necessary to keep up with changing user needs and the operational environment

MERITS OF THE SYSTEM:

- Better data storage

- Faster information retrieval

- Improved accessibility

- Better customer service

LIMITATIONS OF THE SYSTEM:

- It is difficult to interpret handwriting with no distinct separation between characters.
- The strokes, irregularities, spacing of letters and characters, and block or cursive handwriting

FUTURE ENHANCEMENT OF THE PROJECT:

Optical Character Recognition usually abbreviated to OCR involves a computer system designed to translate images of typewritten or handwritten text into machine readable and editable text. OCR could be applied to many fields like vehicle license plate recognition, information retrieval, document digitization, and in text-to-speech applications. Over the years, OCR has attracted a great deal of researches and has developed various successful methods of recognition.

Considering the entire spectrum of industries where need for immediate saving of labor is realized following are some of the future works that can be workedu

CHAPTER 12

CONCLUSION

We have successfully developed Handwritten character recognition (Text Recognition) with Python, TensorFlow, and Machine Learning libraries. Handwritten characters have been recognized with more than 97% test accuracy. This can be also further extended to identifying the handwritten characters of other languages too. Various aspects related to a common OCR engine architecture and its implementation on Tesseract is discussed in-depth. Although, the accuracy of the OCR systems is highly dependable on the quality and nature of the text data, but it has been analyzed that the Tesseract can be proved to be a very efficient OCR system.

CHAPTER 13

BIBLIOGRAPHY

REFERENCES

Python   - Guido van Rossum

HTML   - Ivan Barros

JavaScript   - Ivan Barros

CSS   - Thomas A. Powell

MySQL   - Paul DuBois

WEBSITES VISITED

www.google.com www.w3schools.com www.wikipedia.com

[1] Satish Kumar, Sunil Kumar, Dr. S. Gopinath, "Text Extraction from Images", International Journal of Advanced Research in Computer Engineering & Technology-2012, ISSN: 2278 – 1323, ISSN: 2278 –1323.

[2] Ray Smith, Google Inc., "An Overview of the Tesseract OCR Engine", Proc. of ICDAR 2007, Volume 2, Page(s):629 - 633, 2007.

[3] Sandip Rakshit, Subhadip Basu, Techno India College of Technology, Kolkata, "Development of a multi-user handwriting recognition system using Tesseract open-source OCR engine", Proc. International Conference on C3IT 240-247, 2009.

[4] Chirag Patel, Atul Patel, Dharmendra Patel (CHARUSAT), "Optical Character Recognition by open-source OCR Tool Tesseract: A Case Study", International Journal of Computer Applications (0975 – 8887), Volume 55– No.10, October 2012.

[5] Md. Abul Hasnat, Muttakinur Rahman Chowdhury, Mumit Khan, "An open-source Tesseract based Optical Character Recognizer for Bangla script", 2009 10th International Conference on Document Analysis and Recognition (ICDAR), Barcelona, Spain, 2009, pp. 671-675.

[6] Parag Agrawal, Rohit Varma, "Text Extraction from Images", IJCSET, April 2012, Vol 2, Issue 4,1083-1087.
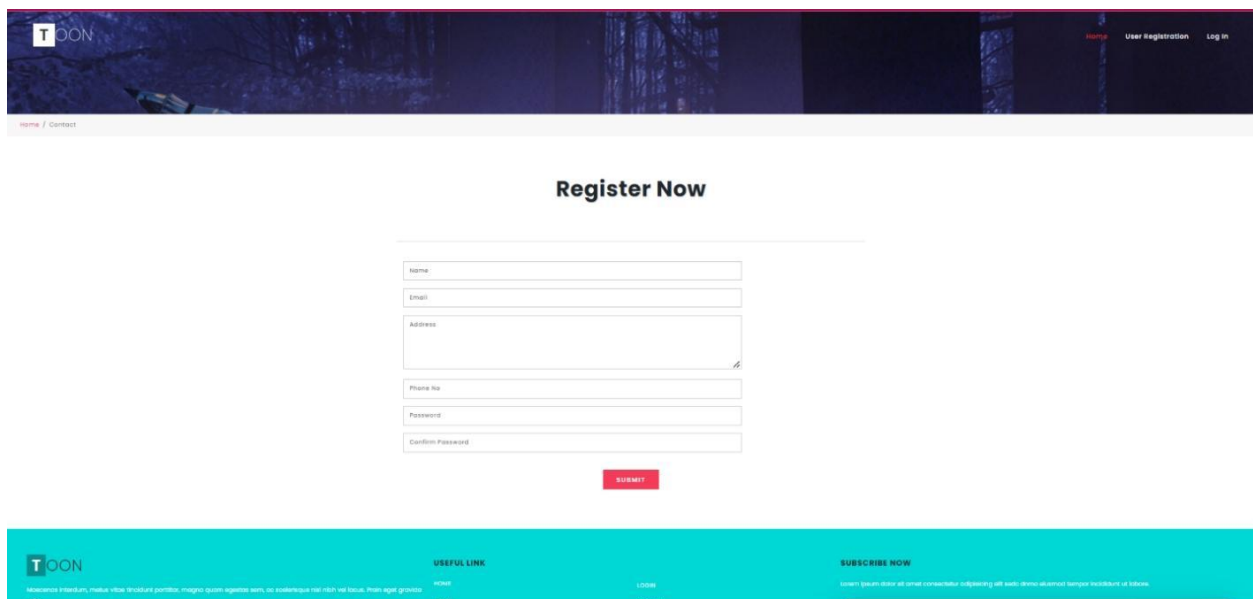
CHAPTER 14

APPENDIX

Front Page

Register Page

Login Page

User Page

TABLE DESIGN

Login

Field	Type	Default	Comments
unnamed	varchar (50)	Not Null	User Name
password	varchar (30)	Not Null	Password
user type	varchar (20)	Not Null	User Type
status	varchar (20)	Not Null	Status

Notes

Field	Type	Default	Comments
nod	int (11)	Primary Key	Note Id
notes	varchar (400)	Not Null	Notes
images	varchar (400)	Not Null	Images
aid	int (11)	Foreign Key	User Id

Userreg

Field	Type	Default	Comments
aid	int (11)	Primary Key	Primary Key
name	varchar (20)	Not Null	Name Of User
email	varchar (40)	Not Null	Email of User
address	varchar (30)	Not Null	Address of User
phone no	varchar (15)	Not Null	Phone Number of User