Image Capture Application

Xoriant Solutions Pvt. Ltd

A training report

Submitted in partial fulfilment of the requirements for the award of degree of

Bachelor of Technology

Computer Science and Engineering

Submitted to

LOVELY PROFESSIONAL UNIVERSITY

PHAGWARA, PUNJAB

From 18/01/21 to 18/05/21

SUBMITTED BY

Name of student: Rahul Rathore

Registration No: 11702112

Signature of Student:

Submitted to:

Name of Supervisor: Suruchi Talwani

Designation: Assistant Professor

Student Declaration

To whom so ever it may concern

I, **Rahul Rathore, 11702112,** hereby declare that the work done by me on "**Image Capture Application**" from **18th January,2021** to **18th May, 2021**, under the supervision of **Mr Deepak Katiyar**, **Technical Architect, Xoriant Solutions Pvt Ltd,** and **Ms. Suruchi Talwani**, **Assistant Professor, Lovely professional University, Phagwara, Punjab**, is a record oforiginal work for the partial fulfilment of the requirements for the award of the degree, **Bachelor of Technology** in **Computer Science and Engineering.**

Rahul Rathore (11702112)

Signature of the student

Dated: 20th May, 2021

Declaration by the Supervisors

To whom so ever it may concern

This is to certify that **Rahul Rathore**, **11702112** from Lovely Professional University, Phagwara, Punjab, has worked as a trainee in **Xoriant Solutions Pvt Ltd** on "**Image Capture Application**" under my supervision from **January, 2021** to **May, 2021**. It is further stated that the work carried out by the student is a record of original work to the best of my knowledge for the partial fulfilment of the requirements for the award of the **Bachelor of Technology** in **Computer Science and Engineering**.

Deepak Katiyar

Technical Architect
(Xoriant Solutions Pvt. Ltd)

Deepak.katiyar@xoriant.com

Dated: 20th May,2021

Suruchi Talwani

Assistant Professor
(Lovely Professional University)

Signature of the Internal Supervisor

Dated:

ACKNOWLEDGEMENT

The internship opportunity I had with Xoriant Solutions Pvt. Ltd was a great chance for learning and professional development. Therefore, I consider myself a fortunate individual as I was provided with an opportunity to be a part of it. I am also grateful for having a chance to meet so many wonderful people and professionals who led me through this internship period.

Bearing in mind previous, I am using this opportunity to express my deepest gratitude and special thanks to my external supervisor, Mr. Deepak Katiyar, Technical Architect (Xoriant Solutions), who, in spite of being busy with his duties, took time out to hear, guide and keep me on the correct path and allowing me to carry out my project under his guidance and supervision.

I express my deepest thanks to my internal supervisor, Ms. Suruchi Talwani, Assistant Professor (Lovely Professional University), for taking part in helpful decisions & giving necessary advice and guidance and arrange all facilities from the University end to make life easier. I choose this moment to acknowledge her contribution gratefully.

I would also like to thank my team at Xoriant Solutions Pvt. Ltd, who extended their kind support and helped towards the completion of this project. It is their help and support, due to which we became able to complete the design and technical report. Without their support, this report would not have been possible.

Rahul Rathore

Registration No.: 11702112

B. Tech – CSE

Lovely Professional University, Phagwara

List of Tables

List of Figures

List of Abbreviations

S No.	Abbreviation	Full-Form
1	ICA	Image Capture Application
2	STS	Spring Tool Suite
3	API	Application Programming Interface
4	OCR	Object Character Recognition

Table of Content

INTRODUCTION OF THE COMPANY

Xoriant is a Silicon Valley headquartered company focused on product engineering, software development, and technology consulting services for technology product companies and innovative enterprises.

From its inception, Xoriant has been a technology leader and execution partner throughout the Build, Run and Transform life cycle for companies that create technology products and enterprises that use those products to implement business solutions. Our customers range from startups to Fortune 100 and encompass BFSI, High Tech, Healthcare, Pharma, Industrial Manufacturing, Telecommunications, and Automotive sectors.

Over three decades, Xoriant and its engineers have employed a rigorous and disciplined approach, resulting in two significant improvements for accelerated product delivery: We have abstracted and captured the most successful methodologies, framework components, accelerators, and technologies for solving critical client challenges. Next, in support of these innovations, we have codified and standardized the engineering methodologies, implementation best practices, measurement methods, and early warning mechanisms while leaving room for continuous enhancements. This is the "X·FACTOR" – the secret sauce of Xoriant – that ensures robustness, extensibility, and continuity for all the technology we deliver.

Founded in 1990 by a 3-person team of industry pioneers, Xoriant has established a global presence as an innovation leader, preferred technology partner, and one of the Best Workplaces by Great Place to Work. Xoriant's 30th anniversary was a significant milestone, underscoring its mission to continue providing exceptional product engineering solutions and services to technology creators and their enterprise customers using technology to solve business problems. Our vision, however, goes above and beyond traditional engineering to explore, test, and recommend future-ready frameworks and approaches that help organizations achieve modernization across their Build, Run, and Transform technology lifecycles.

INTRODUCTION OF THE PROJECT UNDERTAKEN

INTRODUCTION

Image Capture Application intuitively helps transform your paper-based business processes by leveraging best of breed OCR technology. Captures, classifies, and recognizes text. Image Capture Application is a Microsoft Azure based intuitive document digitization solution. It uses Microsoft Azure OCR services to extract printed/ handwritten text images (key-value pairs) from images and multi-page documents. This information can be stored in structured databases and analyzed using various tools for trending, dashboards, and reporting. Sometimes in this document processing we need to process the information. For this document processing we need a software system called IMAGE CAPTURE APPLICATION. This process is also called DOCUMENT IMAGE ANALYSIS (DIA). Thus, our need is to develop character recognition software system to perform Document Image Analysis which transforms documents in paper format to electronic format.

Widely used as a form of data entry from printed paper data records – whether passport documents, invoices, bank statements, computerized receipts, business cards, mail, printouts of static-data, or any suitable documentation. It is a common method of digitizing printed texts so that they can be electronically edited, searched, stored more compactly, displayed on-line, and used in machine processes such as cognitive computing, machine translation, (extracted) text-to-speech, key data and text mining. OCR is a field of research in pattern recognition, artificial intelligence and computer vision.

We start with the image of a document and the image is segmented corresponding to characters and symbols by the initial segmentation process. Then the initial hypotheses for each sub image are generated based on the features extracted from these sub images. These are composed into words which are varied and corrected if necessary.

- **Features of Image Capture Application:**
  i. Upload up to 1000 Pancard documents.
  ii. Reduced time and manual effort.
  iii. Data extraction and storage easy.
  iv. High Accuracy and Fast processing speed.
  v. Error minimized.
  vi. Highly Scalable and Robust Application.

ICA is having two major Interfaces, One for the client and another one for the Admin.

- **Client Side : Client will be authenticated and given privileges for the application given below:**
  i. Upload Pancard documents
  ii. Visualize data in tabular format.
  iii. View Batch Reports
  iv. View Batch Processes
  v. View Pancard Details

On the other hand, admin can grant the service according to the Client's need, proper maintenance and monitoring will be done by the admin.

OBJECTIVE

The main idea behind the development of ICA is to ease the process of storing documentations in digital format. The year old system of manual data entry is hectic, time consuming and prone to error at every stage, thus needs a change. By developing software like ICA, we believe we'll be able to cut the manual intervention by 95% thus making the whole process automated which will eventually increase the productivity of the system. By productivity, we mean the number of documents it can process in an hour compared to the documents processed by a normal data entry person in the same time.

Scope of the Work (Application can be enhanced by implementing below features)

ICA is mainly based upon the OCR system. OCR engines have been developed into many kinds of domain-specific OCR applications, such as receipt OCR, invoice OCR, check OCR, legal billing document OCR.

i. **Font Independent OCR:** Development of OCR considering the multiple font style needs to be developed in the future. The corner point approach is very much useful for the font independent OCR, because, for font or character size, it finds the block and the blocks are analysed to recognize the character.
ii. **OCR for all Indian Languages:** Development of OCR for languages other than English needs to be researched on and developed in the future. The corner point approach is very much useful for the OCR of languages other than English, because, for font or character size, it finds the block and the blocks are analysed to recognize the character.
iii. **Cursive Characters OCR:** There is heavy demand for an OCR system which recognizes handwritten cursive scripts. This avoids keyboard typing and font coding for the image. This method

helps in detecting handwritten characters with a precision of about 90%.

iv. **Speech recognition from OCR:** Speech recognition is one of the most important application today. The recognized Printed or Handwritten OCR could be recorded, and speech output could be generated. This would help the blind to send and receive information.

v. **Speech to text converter through OCR:** Speech recognition is one of the most important application today. The recognized speech could be recorded, and output of text could be generated.

Application

The proposed software system is highly robust and scalable piece of work and can be used at many places with a little or no changes as per the client's need. The expected industries where this system can be implemented includes:

- Data entry for business documents, e.g., Cheque, Passport, Invoice, etc.
- In Schools and Colleges to digitize existing student records and new records on admission.
- In Healthcare sectors to digitize the Patient's records.
- In Insurance Firms for documents key information extraction.
- Extracting business card information into a contact list.
- More quickly make textual versions of printed documents.
- In businesses which require collection of physical copies of User Identity.

Importance and Applicability

- **Searchability:** OCR engine processed data are transferred in searchable text formats, such as Doc, RTF, txt, pdf, etc. Moreover, the required information can be internally found using a computer command Ctrl+F / Command+F.

- **Editability:** Once a paper document is digitized with an character recognition app, it can be easily amended with a word processor.

- **Storability:** Digital text data, both current and historical, are automatically stored in electronic libraries and storages.
- **Accessibility:** Digital storages of electronic documents can be easily accessed by locally distributed users.

- **Translatability:** OCR-powered solutions also embrace translational capabilities. An OCR engine is used for extracting the text from a scanned document and then translating it to a different language.
- **Backups:** Digital back-ups are cost-effective, and potentially, can be done an infinite number of times. Instead of maintaining costly duplicates and triplicates in paper form, object recognition software automatically creates **OCR backups.**

➢ Role and profile

My work profile at Xoriant was of an Intern. I was working directly under the Technical Architect doing majority of work in Python, developing APIs and coordinating with java team to integrate the developed APIs into the main modules and add other features as per demand and requirements. Also final testing and verification of smooth working of the integrated model was done by me along with suggesting the scope of improvements in the developed system.

BREIF INTRODUCTION OF THE PROJECT

System Analysis

Information flow Representation

- **Data Flow Diagram**

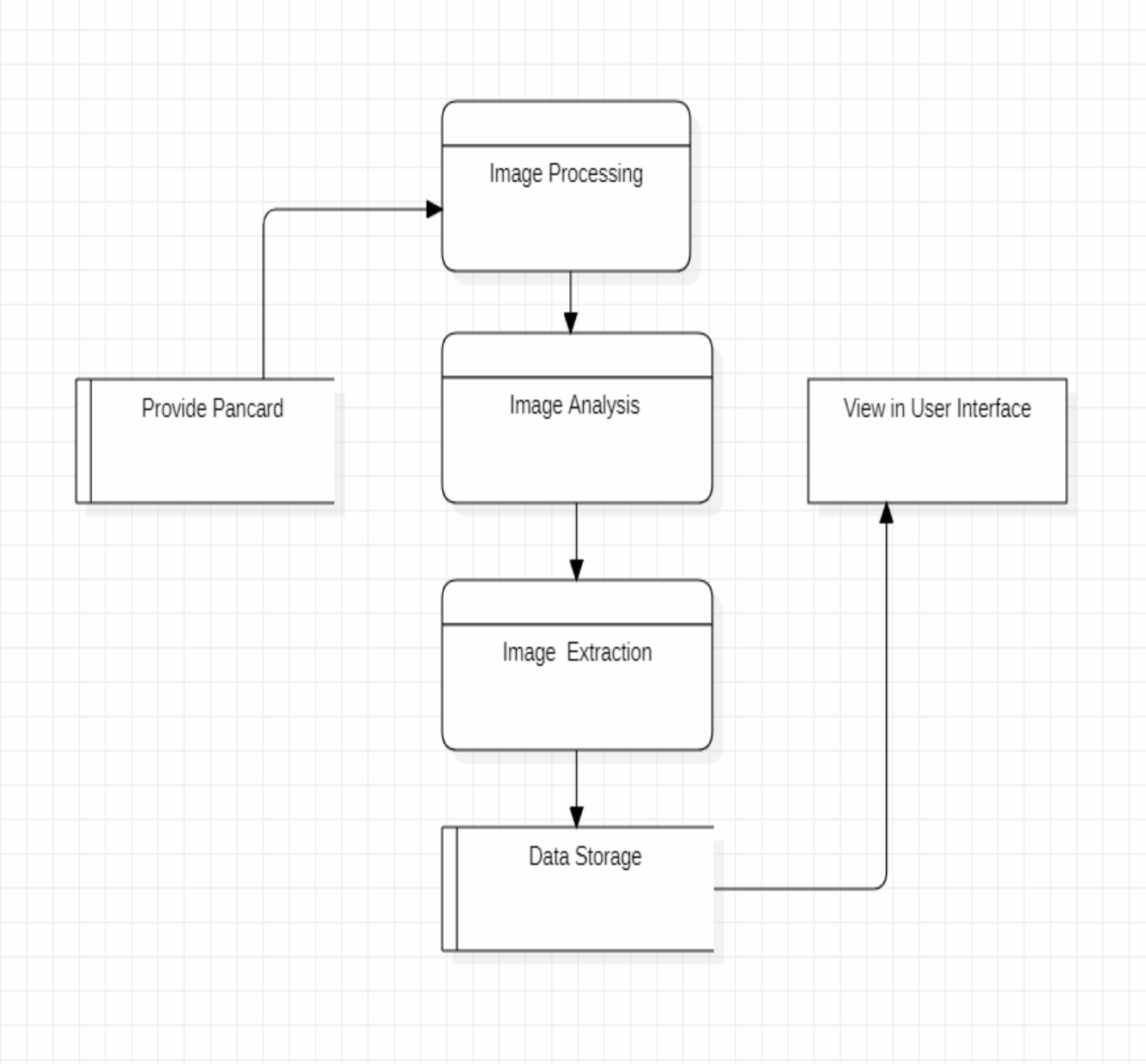

Figure 3. 1

- **Activity Diagram**

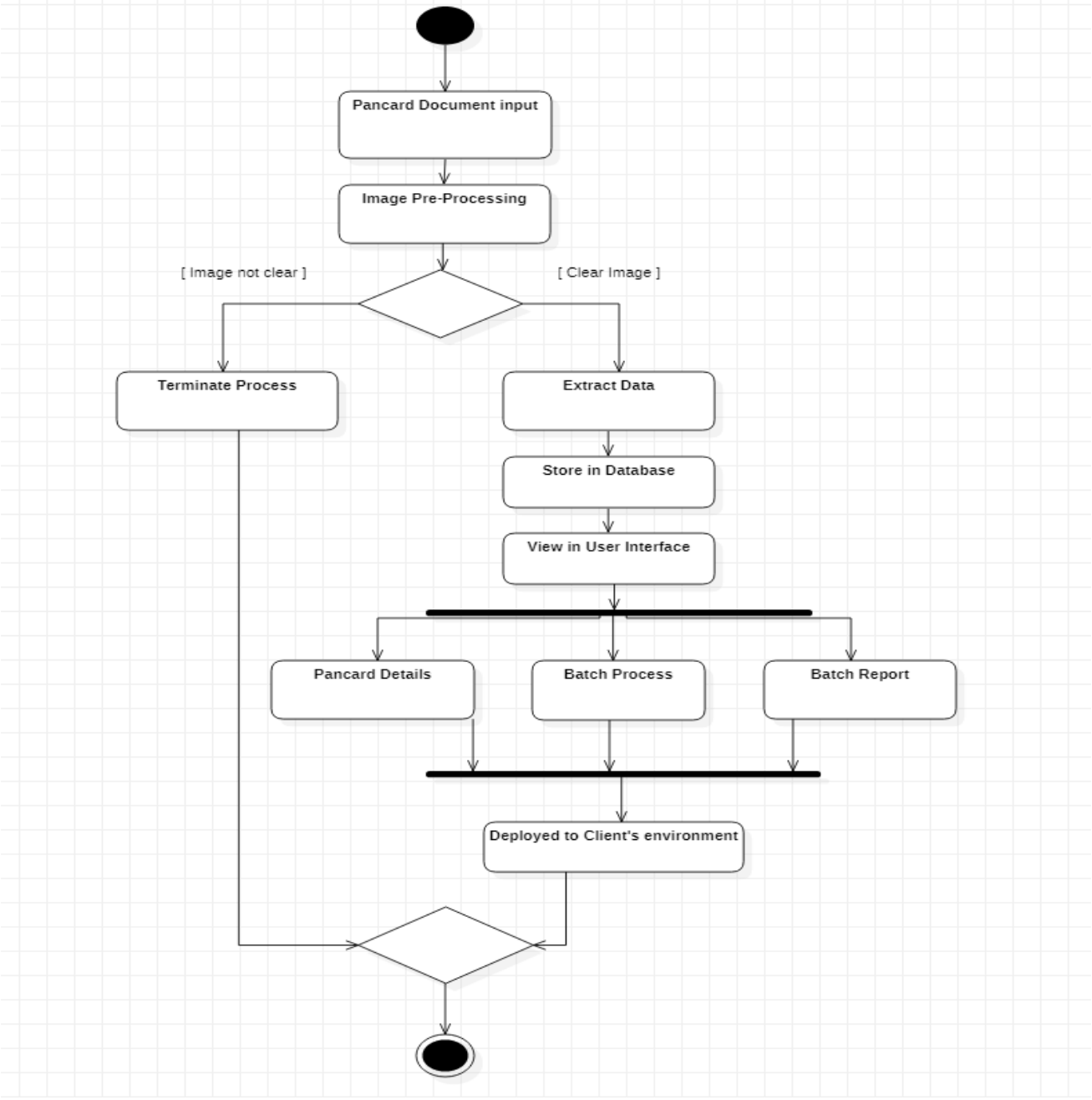

Figure 3. 2

- **Use Case Diagram**

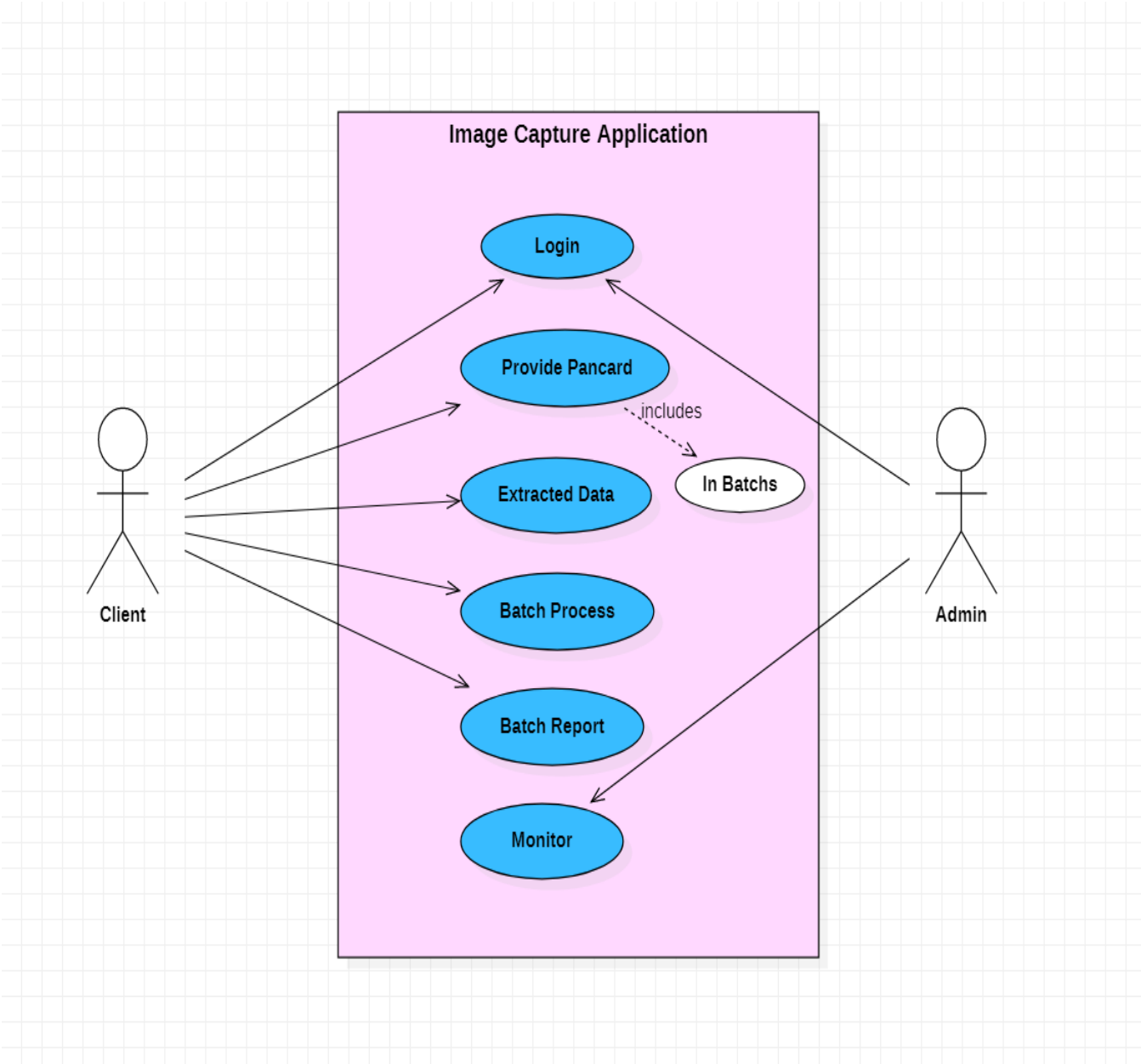

Figure 3. 3

- **Flow Chart**

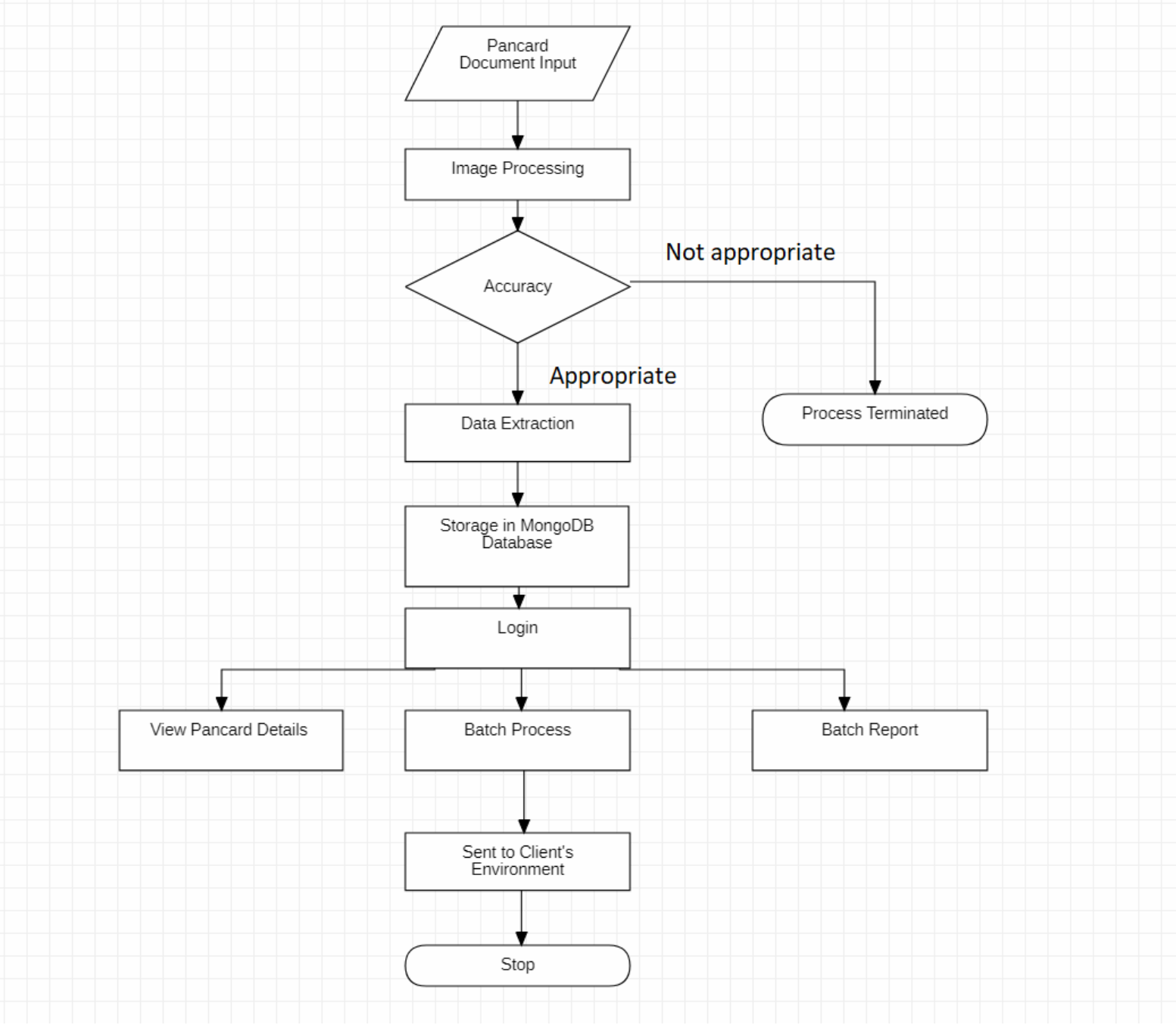

Figure 3. 4

Architectural Design

The software needs the architectural design to represent the design of software. IEEE defines architectural design as “the process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system.” The software that is built for computer-based systems can exhibit one of these many architectural styles.

Each style will describe a system category that consists of:

- A set of components (e.g.: a database, computational modules) that will perform a function required by the system.
- The set of connectors will help in coordination, communication, and cooperation between the components.
- Conditions that how components can be integrated to form the system.
- Semantic models that help the designer to understand the overall properties of the system.

The use of architectural styles is to establish a structure for all the components of the system.

Architectural Behavioural Diagram

Behavioral Diagrams depict the elements of a system that are dependent on time and that convey the dynamic concepts of the system and how they relate to each other. The elements in these diagrams resemble the verbs in a natural language and the relationships that connect them typically convey the passage of time.

Below Diagram depicts the Architecture of ICA and how the supposed functionality will relate to different modules.

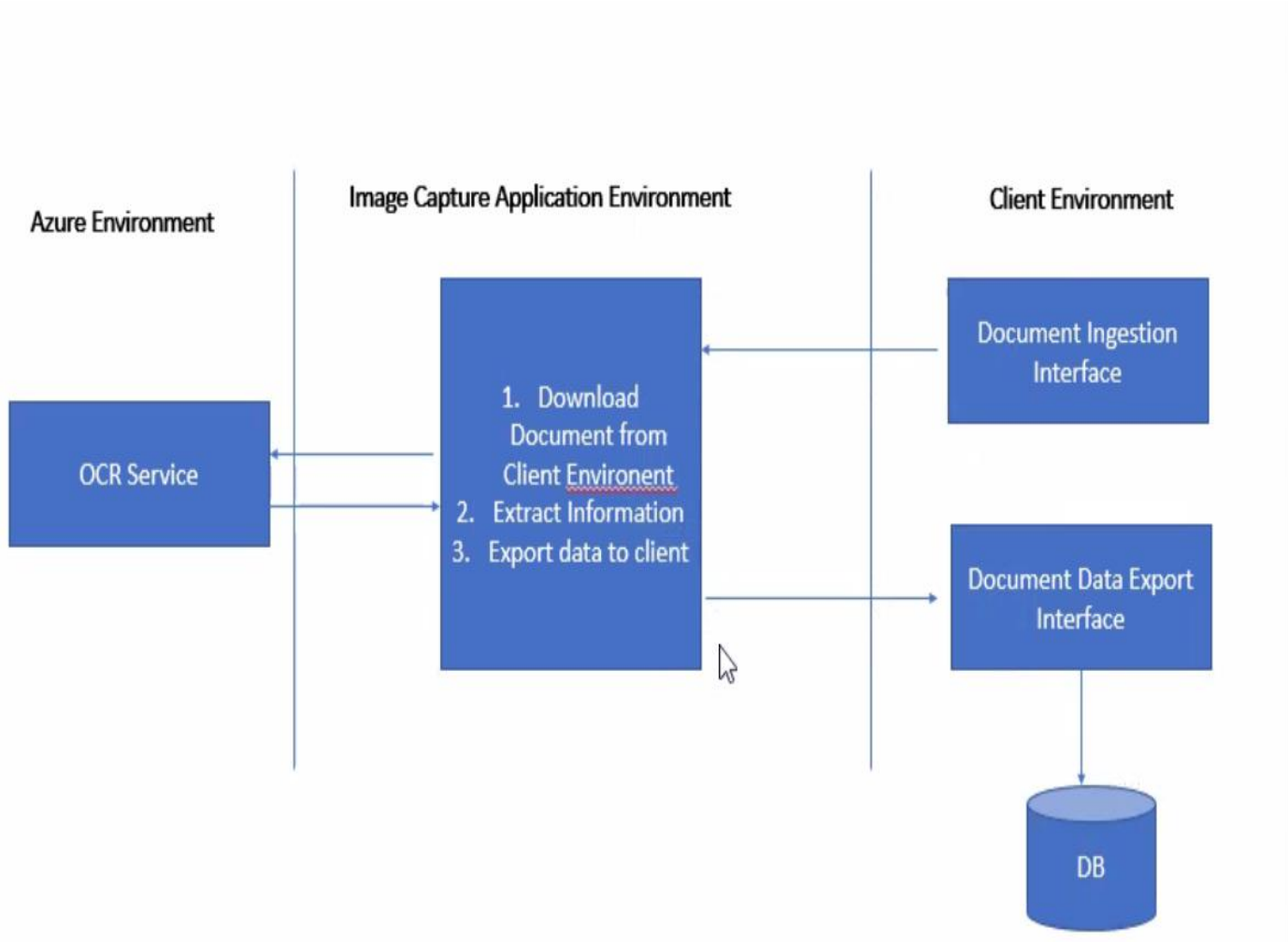

Figure 3. 5

Modular Approach

Modular Approach in Programming Modular programming is the process of subdividing a computer program into separate subprograms. A module is a separate software component. It can often be used in a variety of applications and functions with other components of the system.

• Some programs might have thousands or millions of lines and to manage such programs it becomes quite difficult as there might be too many of syntax errors or logical errors present in the program, so to manage such type of programs concept of modular programming approached.

• Each sub-module contains something necessary to execute only one aspect of the desired functionality.

• Modular programming emphasis on breaking of large programs into small problems to increase the maintainability, readability of the code and to make the program handy to make any changes in future or to correct the errors.

Modules Used

I. Module 1: Local Storage

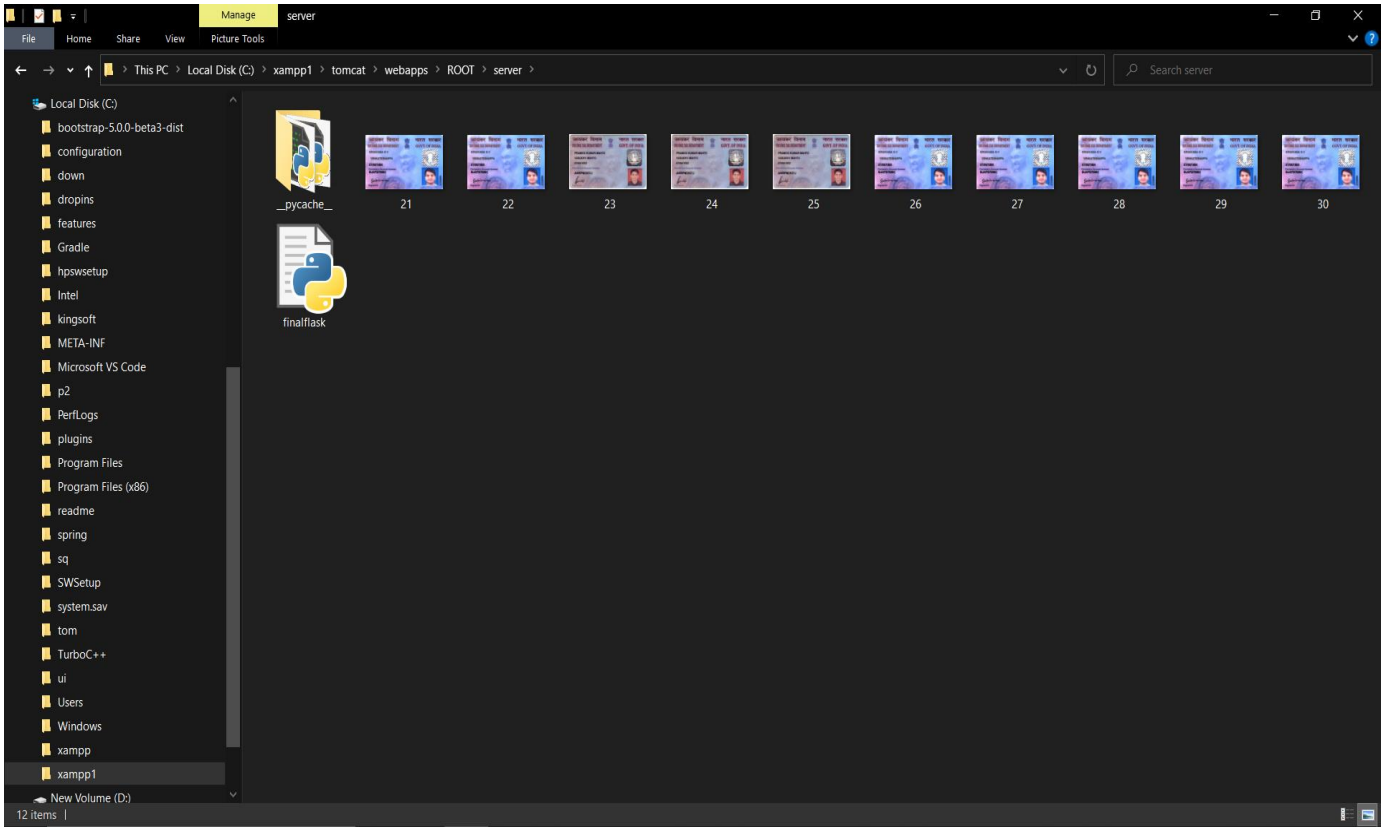

Figure 3. 6

Module 2: Application

- **Fetch files from Local Storage**

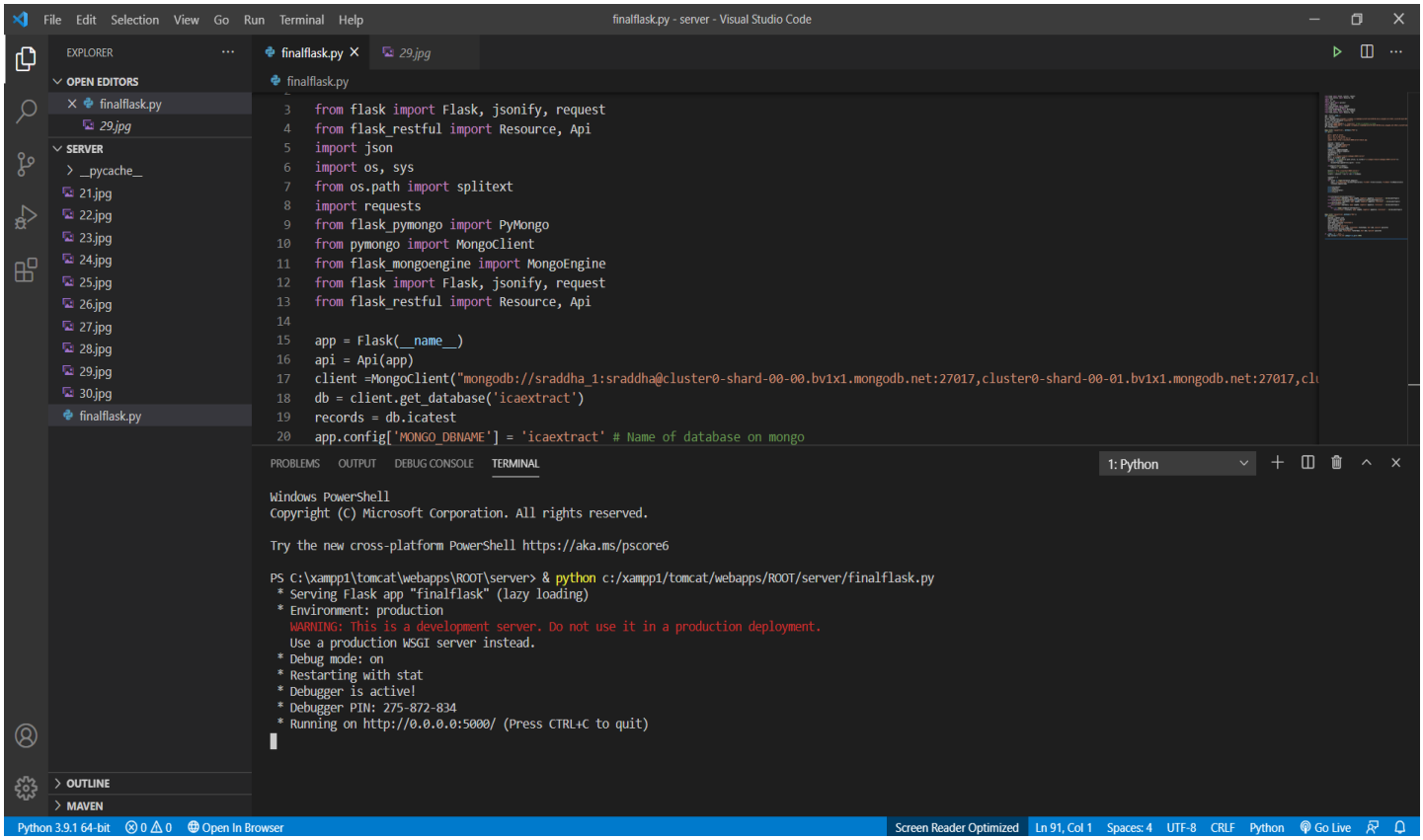

Figure 3. 7

- **Verify the extraction of files from Storage**

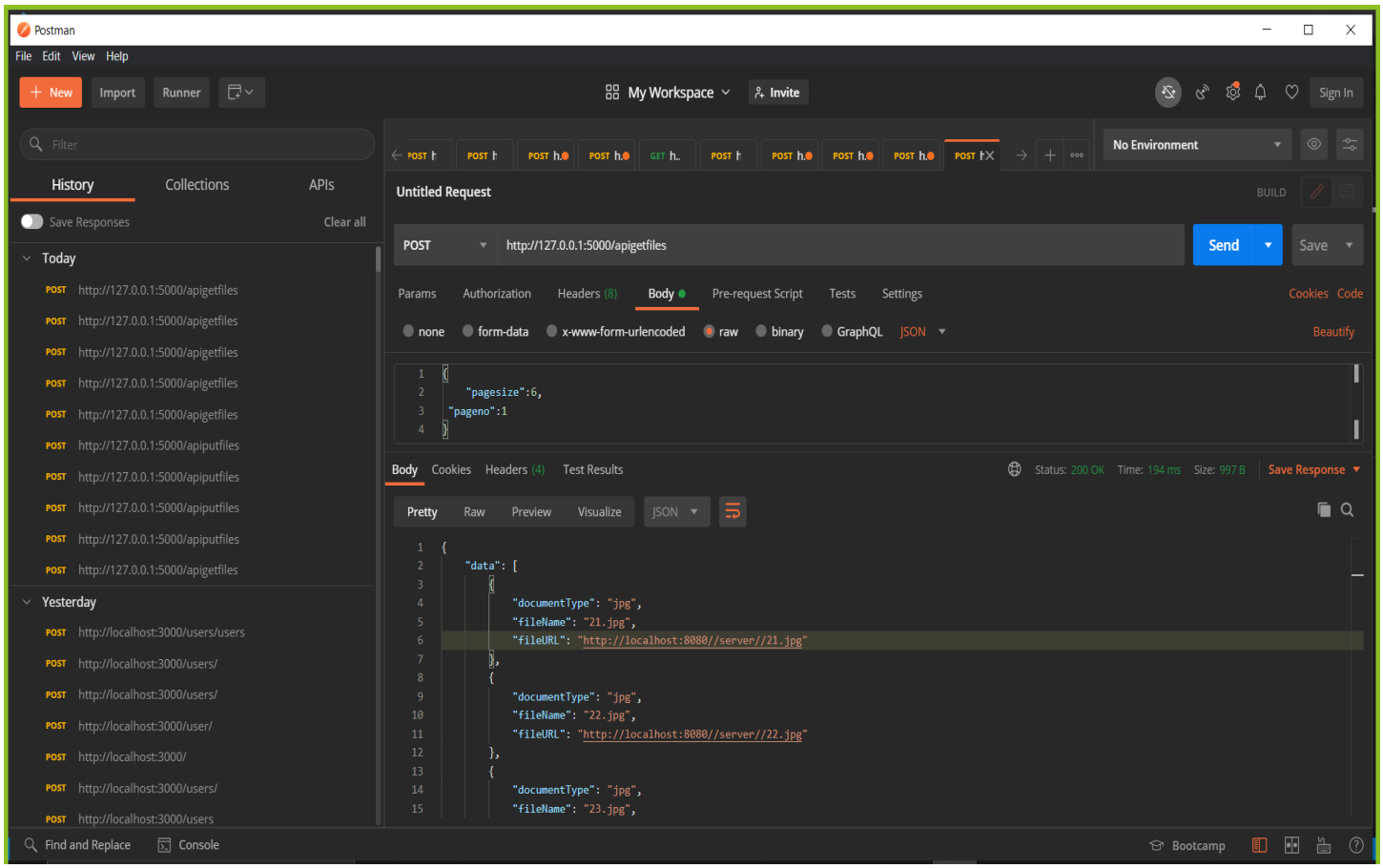

Figure 3. 8

- **Local Server Application**

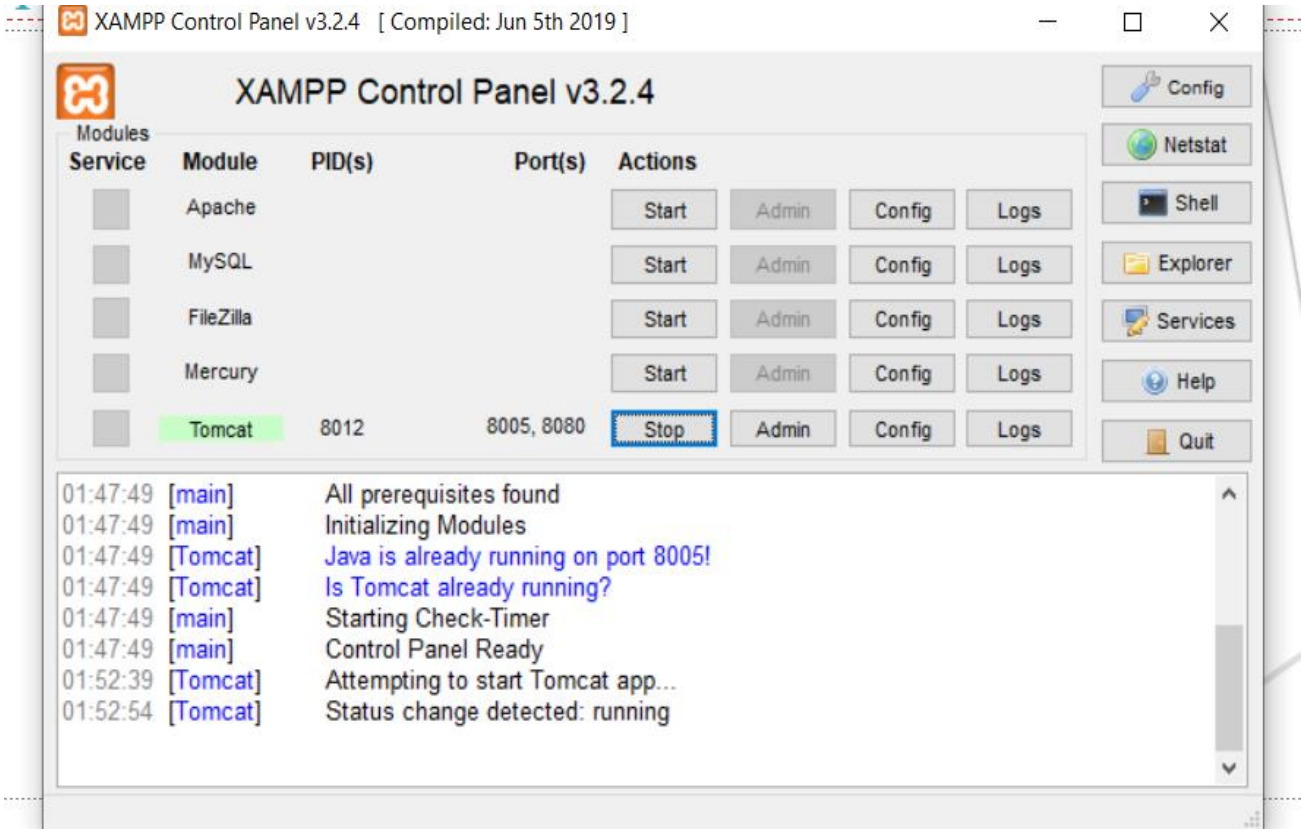

Figure 3. 9

- **Application**

a) The extracted files gets download and stored

b) The OCR is performed and data is extracted

c) The extracted details gets stored in MongoDB atlas

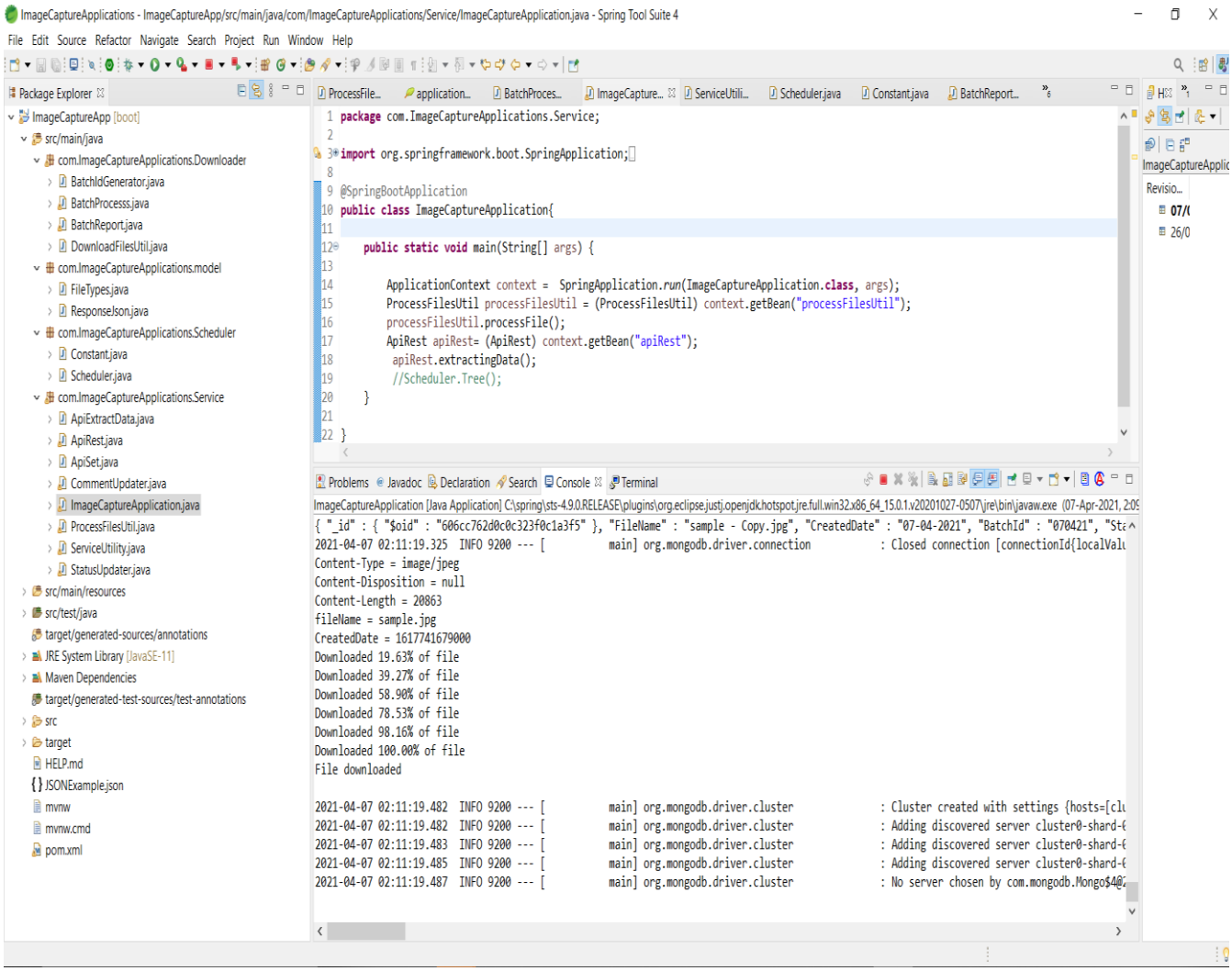

Figure 3. 10

i. Module 3: Cloud Storage

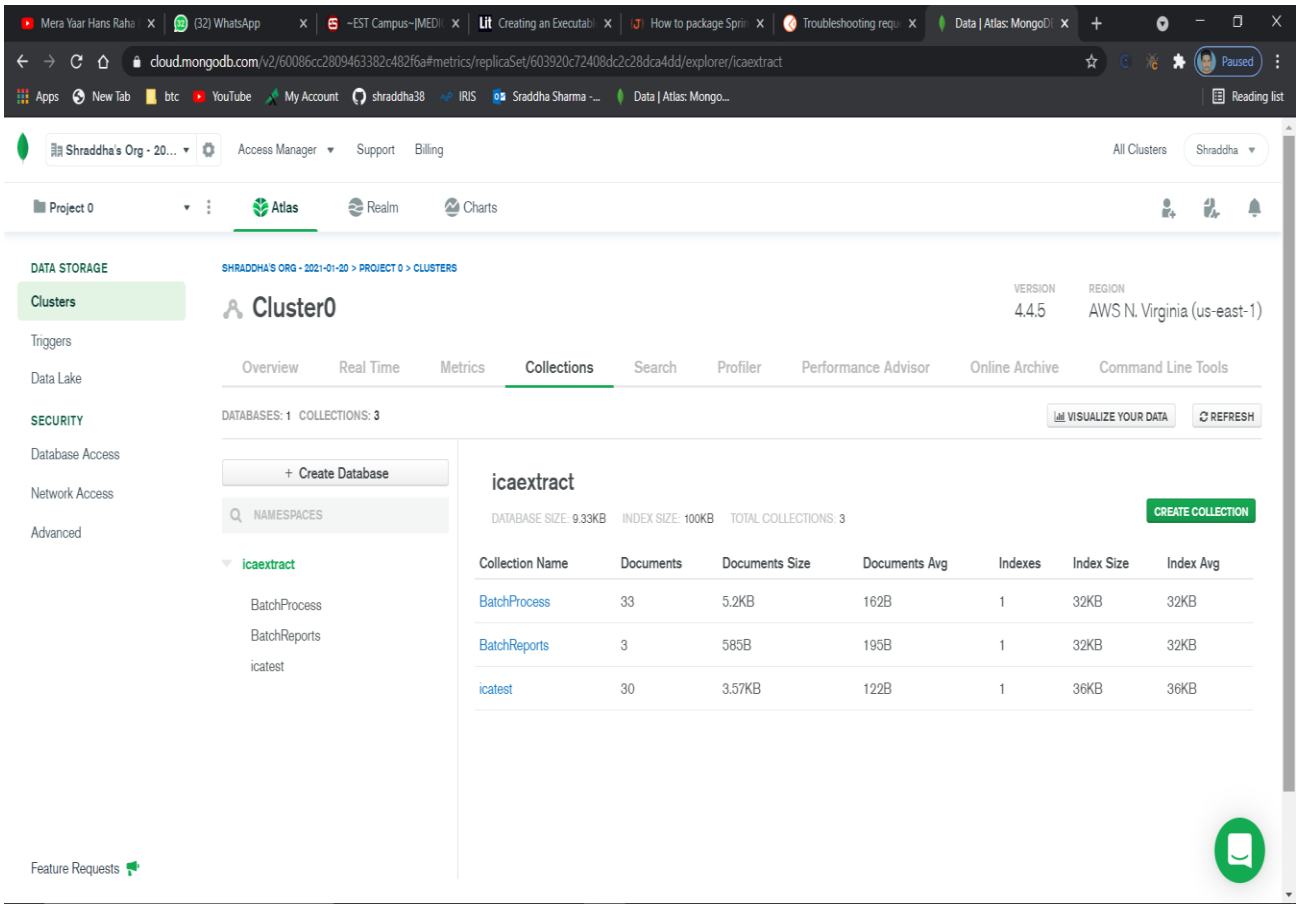

Figure 3. 11

➢ Home Page

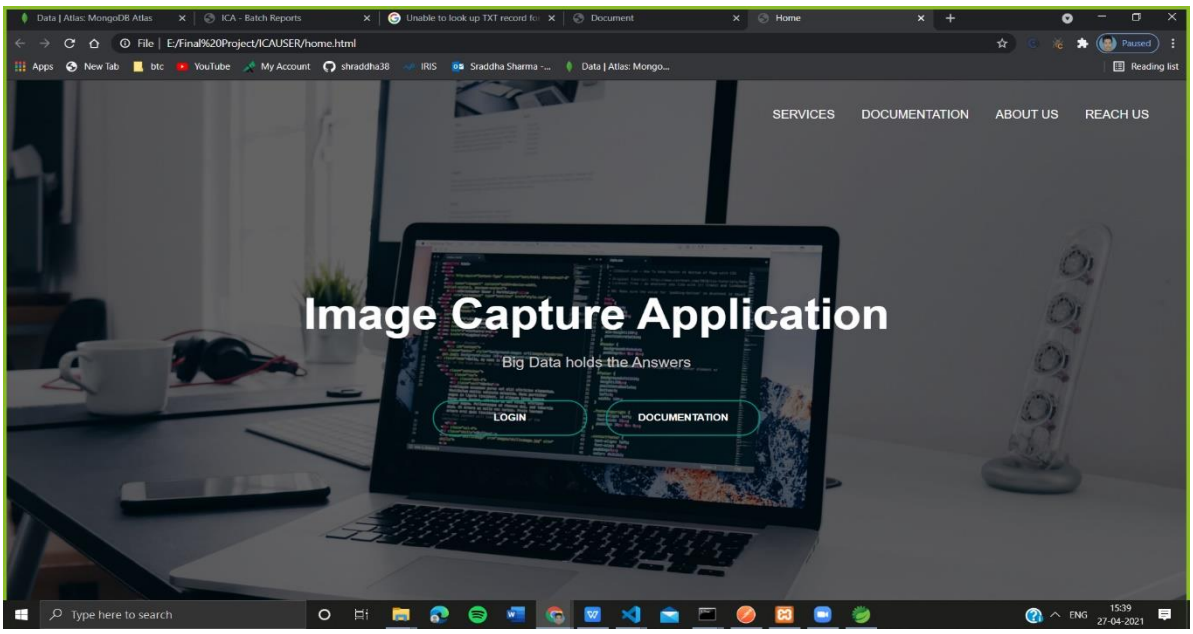

Figure 3. 12

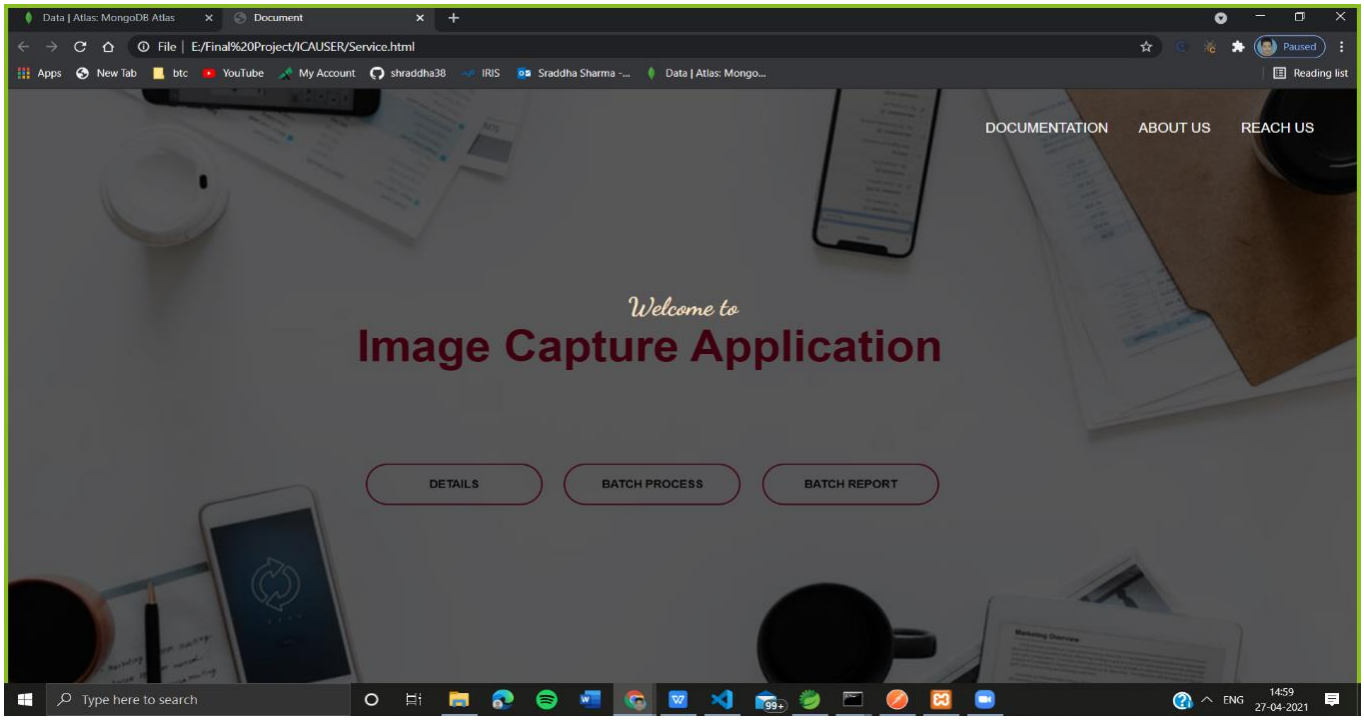

Figure 3. 13

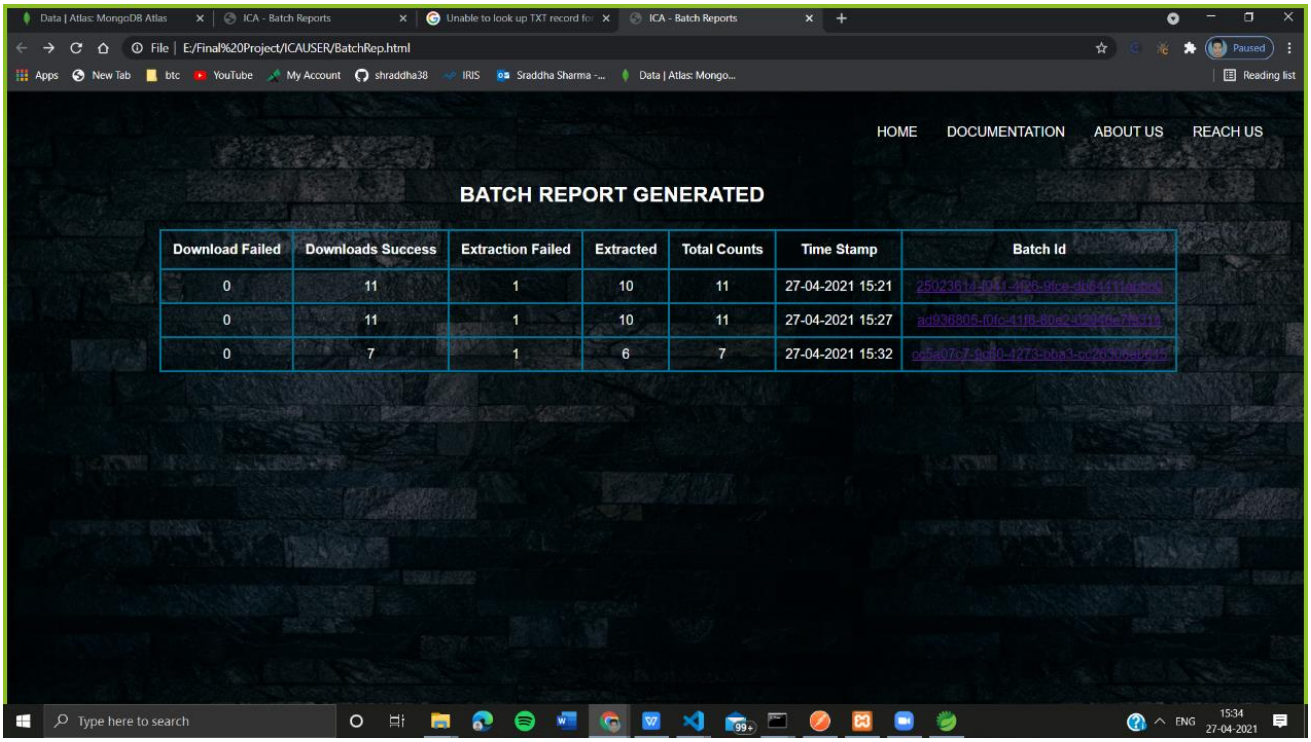

Figure 3. 14

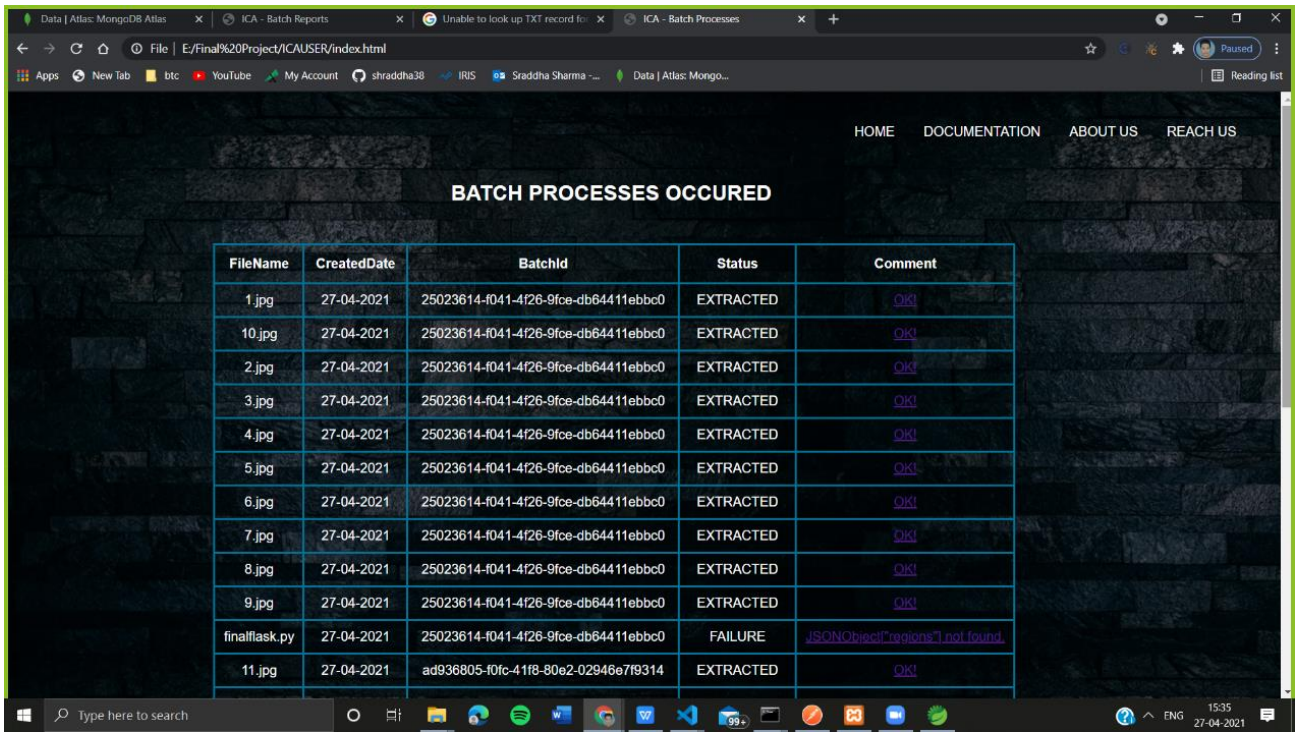

Figure 3. 15

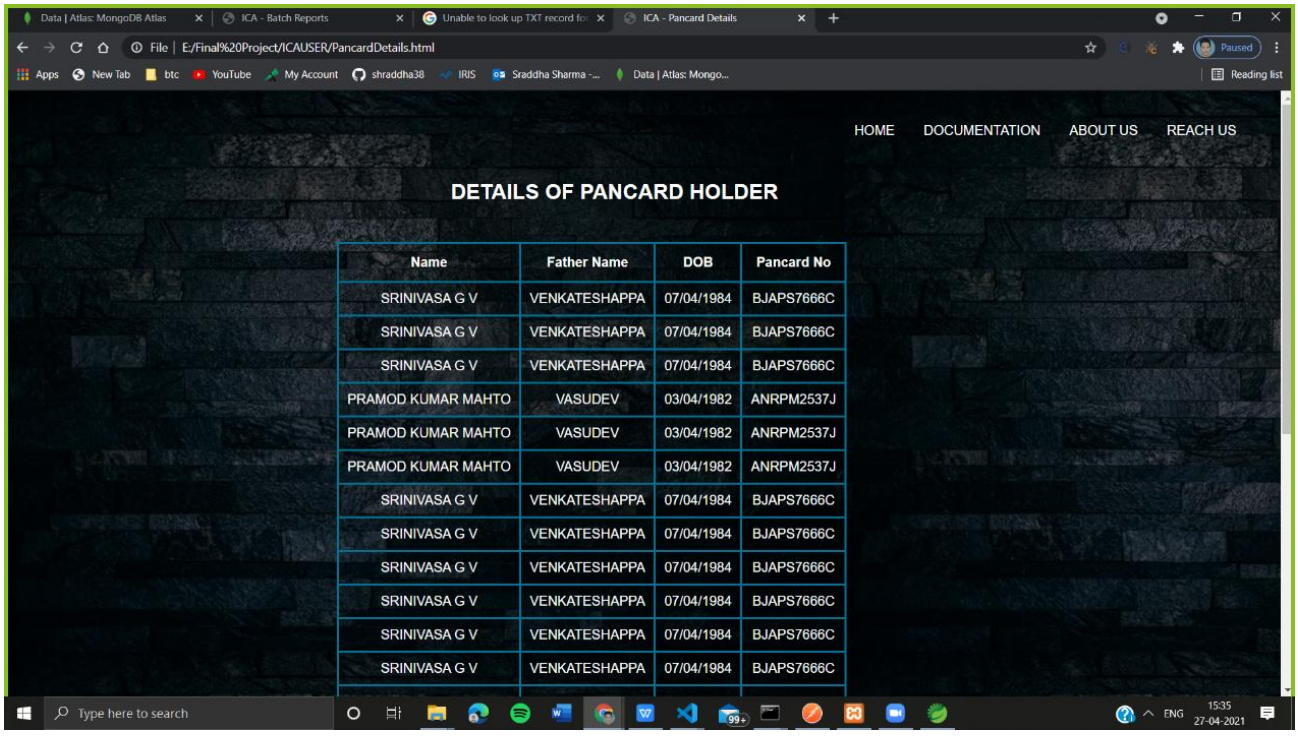

Figure 3. 16

Testing

Software testing is a process of executing a program with the aim of finding the error. To make our software perform well it should be error free. If testing is done successfully, it will remove all the errors from the software.

Testing Objectives

Software testing has different goals and objectives.

The major Objectives of Software testing are as follows:

- Finding defects which may get created by the programmer while developing the software.
- Gaining confidence in and providing information about the level of quality.
- To prevent defects.
- To make sure that the result meets the business and user requirements.
- To ensure that it satisfies the BRS that is Business Requirement Specification and SRS that is System Requirement Specifications.
- To gain the confidence of the customers by providing them a quality product.

Testing Scope

The following things should be taken for testing:

1. The username should be unique to avoid duplicate users.
2. The username and password entered should match the registered user.
3. There should be some files in the local storage before application runs.
4. The file type must be JPEG/PNG for data extraction and storage.
5. The image should be clear and readable.

Testing Principles

Software testing is a process of executing a program with the aim of finding the error. To make our software perform well it should be error free. If testing is done successfully, it will remove all the errors from the software.

There are seven principles in software testing:

1. Testing shows presence of defects.
2. Exhaustive testing is not possible.
3. Early testing
4. Defect clustering
5. Pesticide paradox
6. Testing is context dependent.
7. Absence of errors fallacy

- **Testing shows presence of defects:** The goal of software testing is to make the software fail. Software testing reduces the presence of defects. Software testing talks about the presence of defects and does not talk about the absence of defects. Software testing can ensure that defects are present, but it cannot prove that software is defects free. Even multiple testing can never ensure that software is 100% bug-free. Testing can reduce the number of defects but not removes all defects.

- **Exhaustive testing is not possible:** The process of testing the functionality of a software in all possible inputs (valid or invalid) and pre-conditions is known as exhaustive testing. Exhaustive testing is impossible. This means the software can never test at every test cases. It can test only some test cases and assume that software is correct, and it will produce the correct output in every test cases. If the software will test every test case, then it will take more cost, effort, etc. and which is impractical.

- **Early Testing:** To find the defect in the software, early test activity shall be started. The defect detected in early phases of SDLC will be less expensive. For better performance of software, software testing will start at initial phase i.e. testing will perform at the requirement analysis phase.

- **Defect clustering:** In a project, a small number of the module can contain most of the defects. Pareto Principle to software testing state that 80% of software defect comes from 20% of modules.

- **Pesticide paradox:** Repeating the same test cases again and again will not find new bugs. So, it is necessary to review the test cases and add or update test cases to find new bugs.

- **Testing is context dependent:** Testing approach depends on context of software developed. Different types of software need to perform different types of testing. For example, the testing of an e-commerce website is different from the testing of an Android application.

- **Absence of errors fallacy:** If a built software is 99% bug-free but it does not follow the user requirement then it is unusable. It is not only necessary that software is 99% bug-free but it is also mandatory to fulfil all the customer requirements.

Testing Method Used

Testing methodologies are approaches to testing, from unit testing through system testing and beyond. There is no formally recognized body of testing methodologies, and very rarely will you ever find a unified set of definitions. But here are some common methodologies:

- **Unit Testing:** The act of testing software at the most basic (object) level. Generally performed by developers, run in "friend classes" with code-level access to read and manipulate objects.
- **Acceptance Testing:** Also known as acceptance tests, build verification tests, basic verification tests, these are rudimentary tests which prove whether a given build is worth deeper testing. The term "smoke test" is a colloquial term -- when machines are built, engineers will power them up and just let them run, looking for smoke as a sign of serious problems.
- **Functional testing-**Functional testing takes a user story or a product feature and tests all of the functionality contained within that feature. For example, in a photo application like Photoshop, functional testing would cover all the functionality contained within a feature like opening files (resolving file paths, determining appropriate format filters, passing the file path off to the filter) as well as handling errors within that functionality.
- **System Testing-**Testing the project as a collective system. For the Photoshop application, an example would be to open a file in a given format, manipulate that file in various ways, and then

output the file. System testing generally combines multiple features into an end- to-end process or scenario.

- **Performance Testing-**Tests an application's performance characteristics, be it file size, concurrent users, or mean-time-to-failure.
- **Security Testing-**A collection of tests focused on probing an application's security, or its ability to protect user assets.
- **Black Box Testing-** Black box testing treats the software as a "black box" i.e., without any knowledge of internal implementation. Black box testing methods includes:
    i. Equivalence Partitioning.
    ii. Boundary Value analysis.
    iii. All pair testing.
    iv. Model- Based testing.
    v. Exploratory testing.
    vi. Specification Based testing.

- **White Box Testing -** White box testing is when the tester has access to the internal data structures and algorithms including the code that implement these. White-box testing can be applied at the unit, integration and system levels of the software testing process.

Test Cases

Case 1: Login and Registration

S. No	Input Description	Expected Output	Pass/Fail
1	If username entered is valid	Successful	Pass
2	If password entered is valid	Successful	Pass
3	If Email entered is already in use	Error	Pass

Table 3. 1

Case 2: Application

S. No	Input Description	Expected Output	Pass/Fail
1	If No file is present in the Local Storage	No File Found	Pass
2	If there are 20 files or less	Extracted and Stored	Pass
3	If the file other than type of JPEG/PNG	Extraction Failed	Pass
4	If the image is Blurred	Extracted and Stored	Fail

Table 3. 2

Sample Test Data & Results

Test Case : If the file other than type of JPEG/PNG

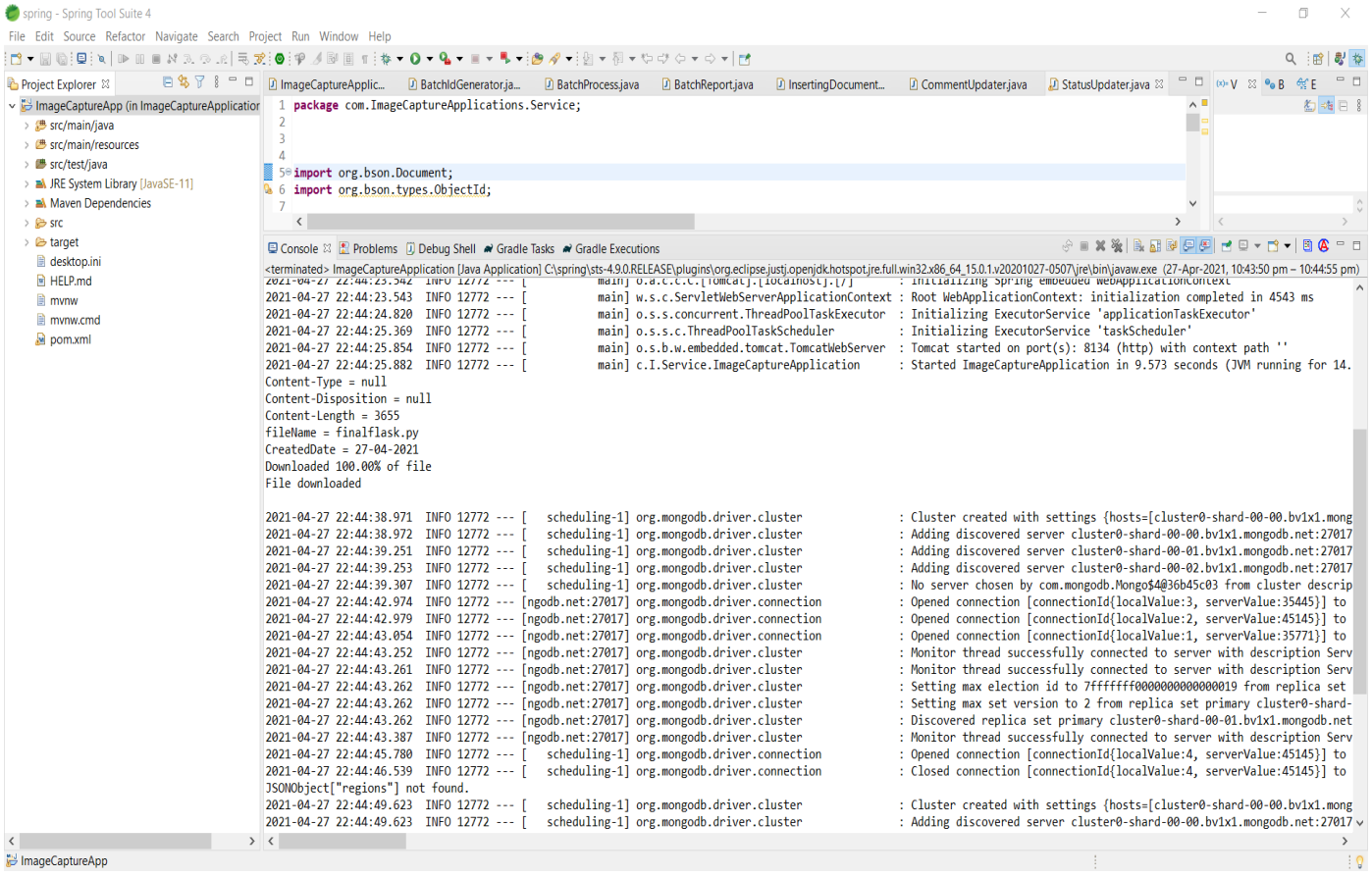

Figure 3. 17

➢ Test Case: If the image is Blurred

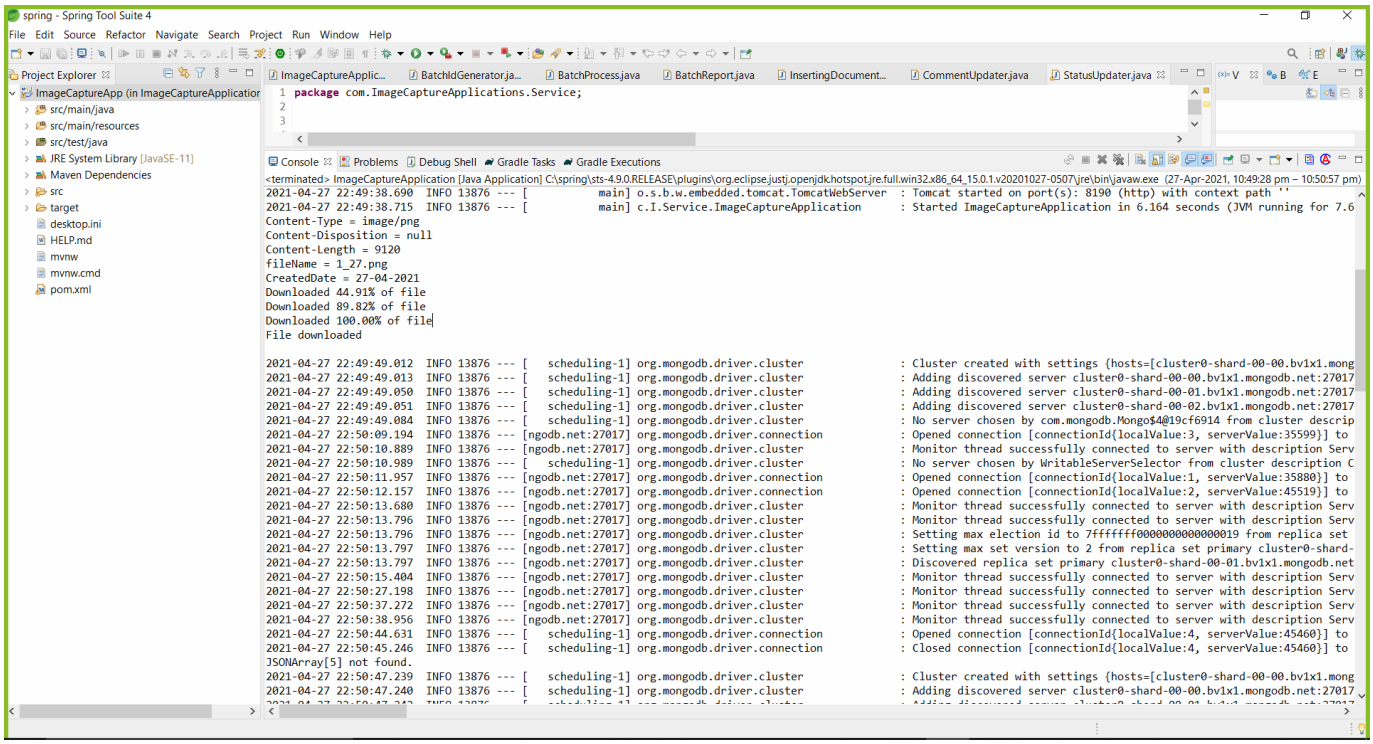

Figure 3. 18

➢ Test Case: If No file is present in the Local Storage


```
_id: ObjectId("6088494607b078061030f58f")
BatchId: "f83871d0-034f-4f5f-9bd7-bf105ae73377"
DownloadFailed: 0
Downloads: 0
ExtractionFailed: 0
Extracted: 0
CreatedDate: "27-04-2021 22:56"
TotalCounts: 0
```


Figure 3. 19

➢ **Email not registered**

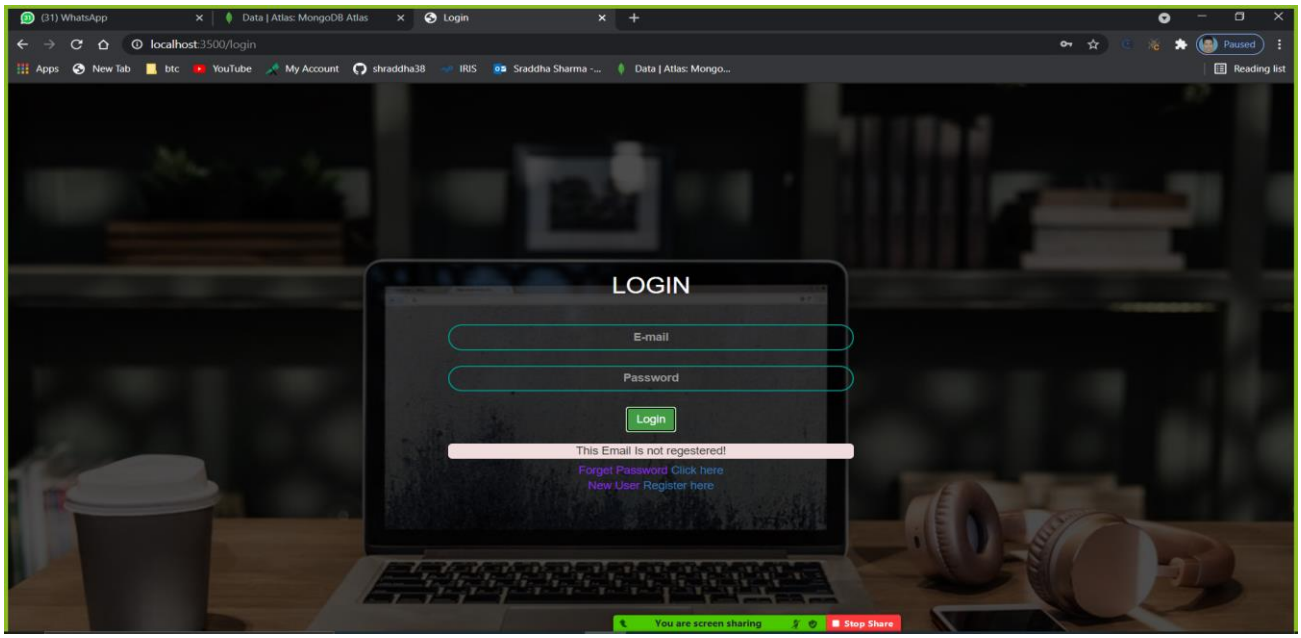

Figure 3. 20

Activities/ Equipment handled

➢ Software

a) Visual Studio Code

Visual Studio Code is a freeware source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

The daily day to day task of creating APIs was done using Visual Studio Code.

b) Spring Tool Suite

Spring Tools 4 is the next generation of Spring tooling for your favourite coding environment. Largely rebuilt from scratch, it provides world-class support for developing Spring-based enterprise applications,

whether you prefer Eclipse, Visual Studio Code, or Theia IDE. Spring Tool Suite is an IDE to develop Spring applications. It is an Eclipse-based development environment. It provides a ready-to-use environment to implement, run, deploy, and debug the application. It validates our application and provides quick fixes for the applications.

The java part of the ICA is completely made using the Spring Tool Suite.

c) Postman

Postman is one of the most popular software testing tools which is used for API testing. With the help of this tool, developers can easily create, test, share, and document APIs.

- Postman is a standalone software testing API (Application Programming Interface) platform to build, test, design, modify, and document APIs. It is a simple Graphic User Interface for sending and viewing HTTP requests and responses.
- While using Postman, for testing purposes, one doesn't need to write any HTTP client network code. Instead, we build test suites called collections and let Postman interact with the API.
- In this tool, nearly any functionality that any developer may need is embedded. This tool has the ability to make various types of HTTP requests like GET, POST, PUT, PATCH, and convert the API to code for languages like JavaScript and Python.

Postman was used to test the APIs before merging them and was used intensively in the testing part to check the boundary conditions.

d) XAMPP Server

XAMPP is an abbreviation where X stands for Cross-Platform, A stands for Apache, M stands for MYSQL, and the Ps stand for PHP and Perl, respectively. It is an open-source package of web solutions that includes Apache distribution for many servers and command-line executables along with modules such as Apache server, MariaDB, PHP, and Perl.

XAMPP helps a local host or server to test its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself.

XAMPP Server was used to create the client's environment locally on our systems and completely customize the ICA as per the needs.

➢ Services Used

a) Microsoft Azure OCR

The cloud-based Computer Vision API provides developers with access to advanced algorithms for processing images and returning information. By uploading an image or specifying an image URL, Microsoft Computer Vision algorithms can analyse visual content in different ways based on inputs and user choices.

MS Azure OCR is the backbone of the application as this helps us to process the image text by returning the texts in a bounding boxes after complete analysis of the image.

➢ Cloud Storage

a) MongoDB Atlas

The most innovative cloud database service on the market, with unmatched data distribution and mobility across AWS, Azure, and Google Cloud, built-in automation for resource and workload optimization, and so much more. MongoDB Atlas is a fully-managed cloud database developed by the same people that build MongoDB. Atlas handles all the complexity of deploying, managing, and healing your deployments on the cloud service provider of your choice (AWS, Azure, and GCP).

MongoDB is the cloud database we are using to store the client data and expanding it as per the demand and needs.

➢ Challenges Faced

It is being truly said that when there are challenges, there are learning opportunities. So were in my internship. Everyday was like a new day, a new beginning where I must be ready to face some new challenges and get the most learning out of it.

The most challenging part of my internship was coping up with the fast pace development team as I was all new there and being directly assigned to a project is a great call for learning.

I used YouTube to explore and learn about technologies I'm not handy with, Read documentations to get to know more about the working and learned from my mentor to get my doubts cleared at any instance of the time at the time of implementations.

- **Learning outcomes**

This Internship was a great learning opportunity for me. Starting my professional life with such a reputed organization helped me learn many great things. I can now confidently say that I have good amount of hands on knowledge about various programming languages. Apart from them, I learnt how to make APIs and integrate them and call them from various applications. Use them efficiently to make our codes more reliable and scalable. I got hands on practice of Microsoft Azure OCR Services and learned how to use them to extract textual data from an image and use them through bounding boxes to get useful information. I also learned about RESTful web services and MongoDB Atlas which is a NoSQL cloud database and learned how to scale up our application as per the client need and demands.

CONCLUSION

Future Perspective

With fast pacing and growing businesses, I believe there is certain need for such application which can be handy and make our work smooth and less sophisticated. Software like ICA can help converting the traditional businesses to a complete digital solution. Following are what in my sense could be the possible areas of development :

- **Font Independent OCR:** Development of OCR considering the multiple font style needs to be developed in the future. The corner point approach is very much useful for the font independent OCR, because, for font or character size, it finds the block and the blocks are analyzed to recognize the character.
- **OCR for all Indian Languages:** Development of OCR for languages other than English needs to be researched on and developed in the future. The corner point approach is very much useful for the OCR of languages other than English, because, for font or character size, it finds the block and the blocks are analyzed to recognize the character.
- **Cursive Characters OCR:** There is heavy demand for an OCR system which recognizes handwritten cursive scripts. This avoids keyboard typing and font coding for the image. This method helps in detecting handwritten characters with a precision of about 90%.
- **Speech recognition from OCR:** Speech recognition is one of the most important application today. The recognized Printed or Handwritten OCR could be recorded and speech output could be generated. This would help the blind to send and receive information.
- **Speech to text converter through OCR:** Speech recognition is one of the most important application today. The recognized speech could be recorded and output of text could be generated.

Conclusion

What does the future hold for Image Capture Application? Given enough entrepreneurial designers and sufficient research and development dollars, ICA can become a powerful tool for future data entry process. However, the limited availability of funds in a capital-short environment could restrict the growth of this application. But, given the proper impetus and encouragement, a lot of benefits can be provided by this ICA.

They are:

1. The automated entry of data by ICA is one of the most attractive, labour reducing technology. The recognition of new font characters by the application is very easy and quick.
2. We can edit the information of the documents more conveniently and we can reuse the edited information as and when required.
3. The extension to software other than editing and searching is topic for future works.
4. A wide range of documents can be processed through this ICA.
5. The extracted and stored data can be more efficiently used for further analysis if needed.
6. Minimal error rate can be expected as the human intervention has been reduced.

The Azure Services used in the ICA along with some scheduling algorithms can be efficiently used to speed up the translation of Image based documents into structured documents that are currently easy to discover, search and process.

.

References

1. Azure Cognitive Services (https://docs.microsoft.com/en-us/azure/cognitive-services/computer-vision/overview-ocr#read-api)
2. Spring Tool Suite Documentation (https://spring.io/guides/gs/sts/)
3. Xoriant's Product - Intuitive OCR Document Digitization (https://azuremarketplace.microsoft.com/en-us/marketplace/consulting-services/xoriantcorporation.xoriant_smart_capture)
4. Flask Documentations (https://flask.palletsprojects.com/en/2.0.x/)
5. Research on Text Detection and Recognition Based on OCR Recognition Technology (https://ieeexplore.ieee.org/document/9236870)