DocuScan

#### A PROJECT REPORT

**Submitted By   
Utpal patel**

**University Roll No- 2200290140169**

**Vaibhav Omar**

**University Roll No- 2200290140172**

#### Submitted in partial fulfillment of the Requirements for the Degree of

MASTER OF COMPUTER APPLICATIONS

#### Under the Supervision of

**Dr. Ankit Verma**

(ASSOCIATE PROFESSOR)

### aktu logo à¤à¥ à¤²à¤¿à¤ à¤à¤®à¥à¤ à¤ªà¤°à¤¿à¤£à¤¾à¤®

### Submitted to

**DEPARTMENT OF COMPUTER APPLICATIONS**

#### KIET Group of Institutions, Ghaziabad Uttar Pradesh-201206

**(March 2024)**

# CERTIFICATE

Certified that **Utpal patel 2200290140169 and Vaibhav Omar 2200290140172** has/ have carried out the project work having “**DocuScan**.” (**Mini Project-KCA353**) for **Master of Computer Application** from Dr. A.P.J. Abdul Kalam Technical University (AKTU**)** (formerly UPTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

**Date:**

**Utpal patel (2200290140169)**

**Vaibhav Omar (2200290140172)**

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

**Dr. Ankit Verma Dr. Arun Tripathi**

**Associate Professor Head of Department**

**Department of Computer Applications Department of Computer Applications**

**KIET Group of Institutions, Ghaziabad KIET Group of Institutions, Ghaziabad**

# ABSTRACT

The project **DocuScan** involves the "Automatic Document-to-Text Converter" project is a cutting-edge initiative designed to streamline the transition from physical documents to digital text, significantly enhancing accessibility, searchability, and overall usability of information. In a world where information is increasingly digitized, this project addresses the crucial need to bridge the gap between traditional paper-based documents and the digital realm.

This document outlines the objectives, methodologies, and key components of the project. Our primary goal is to develop a robust and user-friendly system that automates the conversion of various document types, including printed text, handwritten notes, and scanned documents, into editable and searchable digital text. By doing so, we aim to eliminate the inefficiencies and limitations associated with manual data entry and document transcription

# ACKNOWLEDGEMENTS

Success in life is never attained single-handedly. My deepest gratitude goes to my project supervisor, **Dr. Ankit Verma f**or their guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to **Dr. Arun Kumar Tripathi, Professor and Head of Department, Department of Computer Applications**, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions. Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me with moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and

happiness.

**Utpal Patel (2200290140169)**

**Vaibhav Omar (2200290140172)**

# List of Chapters

|  |  |
| --- | --- |
| Certificate………………………………………………………………………………….. | i |
| Abstract……………………………………………………………………………………. | ii |
| Acknowledgements………………………………………………………………………... | iii |
| Table of Content…………………………………………………………………………… | iv |
| Chapter 1 – Introduction…………………………………………………………………… | 01 - 09 |
| 1.1 Project description……………………………………………………………... | 01 - 02 |
| 1.2 Literature Review……………………………………………………………… | 02 - 06 |
| 1.3 Hardware / Software used in Project…………………………………………... | 07 |
| 1.4 Functional Requirements………………………………………………………. | 08 |
| 1.5 Non- Functional Requirements………………………………………………... | 09 |
| Chapter 2 Feasibility Study………………………………………………………………... | 10 - 11 |
| 2.1 Technical feasibility…………………………………………………………… | 10 |
| 2.2 Operational Feasibility………………………………………………………… | 10 |
| 2.3 Behavioral Feasibility…………………………………………………………. | 11 |
| 2.4 Operational Feasibility………………………………………………………… | 11 |
| Chapter 3 Database Design……………………………………………………………….. | 12 - 24 |
| 3.1 Waterfall Model……………………………………………………………….. | 12 |
| 3.2 Requirement Gathering & Analysis…………………………………………… | 13 |
| 3.3 ER Diagram……………………………………………………………………. | 14 - 16 |
| 3.4 Use Case Diagram……………………………………………………………... | 17 |
| 3.5 Activity Diagram………………………………………………………………. | 18 |
| 3.6 Sequential Diagram……………………………………………………………. | 19 - 20 |
| 3.7 Collaboration Diagram………………………………………………………… | 21 |
| 3.8 State Chart Diagram…………………………………………………………… | 22 |
| 3.9 Component Diagram…………………………………………………………... | 23 |

[3.10 Deployment Diagram 24](#_TOC_250002)

Chapter 4 Form Design 25 - 30

4.1 Screenshot 25 - 30

Chapter 5 Coding 31 - 76

5.1 Module wise code 31 - 76

[Chapter 6 Conclusion 77 - 78](#_TOC_250001)

Chapter 7 Future Scope 79

**Chapter – 1 Introduction**

#### 1.1PROJECT DESCRIPTION

In the realm of digitalization, document scanning plays a crucial role in converting physical documents into digital format, facilitating easier storage, retrieval, and sharing of information. This project aims to create a Document Scanner using OpenCV and PyTesseract, two powerful Python libraries widely used in computer vision and optical character recognition (OCR) tasks.

**1. Background:**

With the advent of technology, there is an increasing demand for efficient document processing. The Document Scanner project addresses this need by automating the process of document digitization. OpenCV (Open Source Computer Vision Library) provides tools for image processing and computer vision, while PyTesseract is a Python wrapper for Google's Tesseract-OCR Engine, enabling the extraction of text from images.

**2. Objectives:**

* Develop a system capable of capturing images of documents using a camera or pre-existing images.
* Utilize OpenCV to preprocess the captured images, enhancing their quality for subsequent OCR.
* Implement Optical Character Recognition (OCR) using PyTesseract to extract text content from the scanned documents.
* Integrate the OCR results with the scanned images to create searchable and editable digital documents.
* Enhance the user interface for ease of use, allowing users to interact with the document scanner efficiently.

**3. Key Components:**

**Image Acquisition**: Utilize a camera module or pre-existing images to capture document snapshots.

**Image Preprocessing**: Apply OpenCV functions to enhance image quality, including resizing, cropping, and adjusting contrast.

**Text Extraction:** Employ PyTesseract for OCR to recognize and extract text content from the processed images.

**Document Assembly**: Combine the OCR results with the original images to create digital documents.

**User Interface:** Develop an intuitive interface for users to interact with the document scanning system.

**4. Workflow:**

**Image Capture**: Capture images of documents using a camera or select existing images for processing.

**Preprocessing:** Use OpenCV functions to enhance the quality of images for better OCR accuracy.

**OCR Processing:** Apply PyTesseract to recognize and extract text from the preprocessed images.

**Document Assembly:** Combine the original images with the extracted text to create digital documents.

**User Interaction**: Provide a user-friendly interface for users to control the document scanning process and access the digitized documents.

**5. Benefits:**

* Streamlines the process of document digitization.
* Improves accessibility and searchability of document content.
* Reduces manual effort in transcription and data entry.

# LITERATURE REVIEW

A literature review for a project report on a Document Scanner using OCR, OpenCV, and PyTesseract would typically involve an exploration of existing research, studies, and relevant literature related to document scanning, OCR technology, and the integration of computer vision libraries like OpenCV and PyTesseract. Here's an outline to guide your literature review:

**1. Overview of Document Scanning:**

* Discuss the evolution of document scanning technology.
* Explore the importance of document digitization in various fields.
* Review existing document scanning methods and their limitations.

**2. Optical Character Recognition (OCR) Technology:**

* Provide an overview of OCR technology and its applications.
* Explore the historical development and milestones in OCR.
* Review existing OCR algorithms and techniques.
* Highlight the challenges and advancements in OCR accuracy.

**3. OpenCV in Document Image Processing:**

* Introduce OpenCV and its role in computer vision applications.
* Discuss how OpenCV is commonly used in document image processing.
* Explore relevant OpenCV functions for image enhancement and preprocessing.
* Review studies and projects that have utilized OpenCV for document scanning.

**4. PyTesseract and OCR Integration:**

* Provide an overview of PyTesseract and its connection to the Tesseract OCR engine.
* Discuss the strengths and limitations of PyTesseract.
* Review literature on the integration of PyTesseract with document scanning projects.
* Explore how PyTesseract enhances OCR capabilities and accuracy.

**5. Challenges and Solutions in Document Scanning with OCR:**

* Identify common challenges in document scanning projects using OCR.
* Review existing research on addressing OCR challenges.
* Discuss solutions proposed by researchers to improve OCR accuracy in document scanning.

**6. Case Studies and Projects:**

* Review relevant case studies and projects that have implemented document scanning with OCR.
* Highlight key findings, methodologies, and outcomes of these projects.
* Compare and contrast different approaches used in similar projects.

**7. User Experience and Interface Design:**

* Explore literature related to user interfaces in document scanning applications.
* Discuss studies on user experience and the design of intuitive interfaces for document scanning.
* Identify best practices and recommendations for user interaction in similar projects.

**8. Future Trends and Emerging Technologies:**

* Discuss emerging trends in document scanning, OCR, and computer vision.
* Explore potential advancements in OCR technology and its impact on document digitization.
* Highlight areas for future research and development.

# 1.2 Software Used in Project

* + - Operating System – Windows 10.
    - Code Editor – Jupiter Notebook.
    - Numpy, Pandas,OCR,Open CV,pytesseract
    - Python: for Programming Language
    - Reading Images from inside file

# Hardware Used in Project

* + - Processor – Intel 5 7th Gen 3500H
    - RAM – 16 GB
    - Graphic Card – NVDIA GEFORCE( 2 GB )

# FUNCTIONAL REQUIREMENTS

Functional requirements describe the specific functionalities and features that a system or software application must possess to meet the intended objectives. Here are functional requirements for a Document Scanner project using OCR, OpenCV, and PyTesseract:

**1. Image Capture:**

The system should support image capture through a connected camera or the upload of pre-existing images.

It should provide options for adjusting camera settings, such as resolution and focus, for optimal image quality.

**2. Preprocessing:**

The system must utilize OpenCV to preprocess captured images, including resizing, cropping, and adjusting contrast for better OCR accuracy.

Users should have the option to preview and approve the preprocessed images before OCR processing.

**3. OCR Processing:**

The system should integrate PyTesseract to perform OCR on preprocessed images for text extraction.

Options for language selection and OCR configuration should be provided to enhance recognition accuracy.

Users should be able to review and edit OCR results before finalizing the document.

**4. Document Assembly:**

The system must combine the original images with the extracted text to create digital documents.

Provide options for choosing output formats (e.g., PDF, text files) and organizing the layout of the digitized document.

**5. Multi-Page Document Handling:**

Support scanning and processing of multi-page documents.

Allow users to organize and rearrange pages within a document.

**6. User Interface:**

Implement an intuitive and user-friendly interface for easy interaction.

Include controls for image capture, image preprocessing, OCR processing, and document assembly.

Display progress indicators and notifications during each stage of the scanning process.

**7. Image Editing Features:**

Incorporate basic image editing features, such as rotation and cropping, to further enhance user control over document appearance.

**8. OCR Validation and Correction:**

Provide a mechanism for users to validate and correct OCR results.

Allow manual editing of recognized text and automatic correction suggestions.

**9. Export Options:**

Allow users to export the digitized documents in various formats, such as PDF, plain text, or other commonly used document formats.

Include options for saving documents locally or sharing them through different channels.

**10. Language Support:**

Support multiple languages for OCR processing to accommodate diverse user needs.

**11. Accessibility Features:**

Implement features to enhance accessibility, such as compatibility with screen readers and keyboard shortcuts.

**12. System Configuration:**

Provide settings for configuring system preferences, including default OCR language, image preprocessing parameters, and output format preferences.

**13. Security Measures:**

Implement measures to ensure the security of scanned documents, including encryption options and secure user authentication.

**14. Logging and Reporting:**

Maintain a log of scanning activities, errors, and user interactions.

Include reporting features for generating summaries or detailed reports of scanned documents.

These functional requirements provide a comprehensive basis for developing a Document Scanner using OCR, OpenCV, and PyTesseract, ensuring that the system meets the needs of users while offering a robust and efficient document digitization solution.

# NON-FUNCTIONAL REQUIREMENTS

# Performance:

# Response Time: The system should have a low response time for capturing, processing, and assembling documents to ensure a smooth user experience.

# Throughput: The system should support scanning and processing multiple documents concurrently.

# 2. Reliability:

# The system should be reliable and resilient, with error-handling mechanisms to gracefully handle unexpected situations, preventing data loss or corruption.

# 3. Scalability:

# The system should be scalable to accommodate an increasing number of users, documents, and processing demands.

# 4. Compatibility:

# The application should be compatible with a range of devices (desktops, laptops, tablets) and operating systems (Windows, macOS, Linux).

# 5. Usability:

# The user interface should be intuitive and user-friendly, with clear instructions and feedback during each stage of the document scanning process.

# The system should support different levels of user expertise, from novice to expert.

# 6. Accessibility:

# The application should adhere to accessibility standards to ensure usability for individuals with disabilities.

# Provide support for screen readers and ensure that all functionality is accessible using keyboard controls.

# 7. Security:

# Data Security: Ensure the security of scanned documents by implementing encryption during transmission and storage.

# User Authentication: Implement secure user authentication to control access to sensitive functionalities and scanned documents.

# Chapter-2

### FEASIBILITY STUDY

After studying and analyzing all the existing and required functionalities of the system, the next task is to do the feasibility study for the project. The feasibility study includes consideration of all the possible ways to provide a solution to a given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily made based on the future upcoming requirements.

### Technical Feasibility

This included the study of function, performance and constraints that may affect the ability to achieve an acceptable system. For this feasibility study, we studied complete functionalities to be provided in the system, as described in the System Requirement Specification (SRS) and checked if everything was possible using different types of front end and backend platform.

### Operational Feasibility

Operational feasibility study is an assessment of whether a proposed project is feasible from an operational standpoint. It evaluates the practicality of implementing the project based on factors such as technology, personnel, infrastructure, and processes. This study aims to identify potential operational issues that may arise during the project's implementation and determine whether the project can be realistically executed within the available resources and constraints. The operational feasibility study is an essential step in project planning, as it helps ensure that the project will be successful and sustainable over the long term.

### Behavioral Feasibility

A behavioral feasibility study is an examination of the potential impacts that a project or initiative may have on human behavior. It assesses whether the project aligns with societal norms and values, and whether it is likely to be embraced by the community it is intended to serve. This type of study explores how the target audience may react to the project and what factors could influence their behavior. It also considers the feasibility of implementing the project, considering the attitudes, beliefs, and cultural values of those who will be affected. The goal of a behavioral feasibility study is to ensure that a project is not only financially feasible, but also socially acceptable and culturally appropriate.

### Economic Feasibility

For the economic feasibility, Economic analysis or cost/benefits analysis is most frequently used technique the effectiveness of a proposed system. It is a procedure to determine the benefits and saving those that are expected from the proposes system and compare them with cost. If the benefits outweigh the costs, a decision is taken to design and implement the system. otherwise, further justification or alternative in proposed system will have to be made if it is to have a chance of being approved this is ongoing effort that improves in accuracy at each phase of a system life cycle.

# Chapter-3

### REQUIREMENTS GATHERING & ANALYSIS

In this Phase, all possible requirements of the system are captured and documented in a requirement specification doc.

### System Design:

The requirements documented in the previous phase are studied in this phase and the system design is prepared.

### Implementation:

With inputs from system design, the system is developed in several units. Then the units are tested.

### Integration & Testing:

The units of the program developed in the previous phase are integrated into a system. Then the whole system is tested.

### Deployment of the system:

When all kinds of testing is done, the product is deployed in the customer environment.

### Maintenance:

There are some issues which are found in the client environment. Patches are released to fix those issues.

# USE CASE DIAGRAM

Use-case diagrams model the behavior of a system and help to capture the requirements of the system. Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors.

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and tells how the user handles a system.

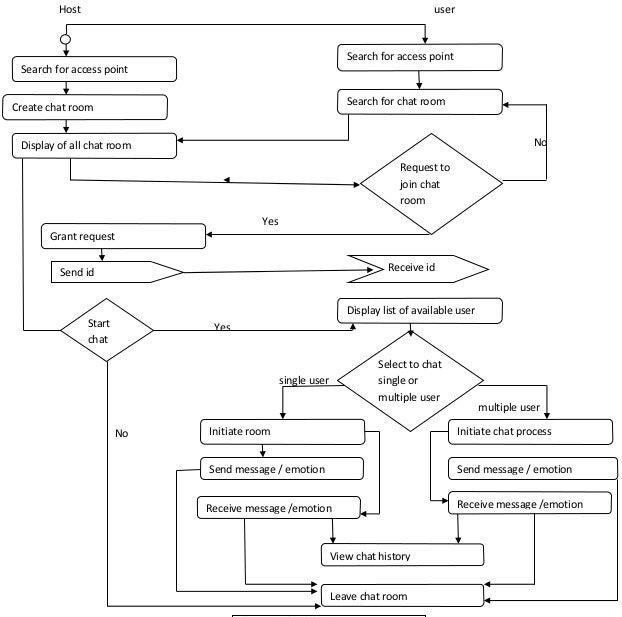
Purposes of a use case diagram given below:

1. It gathers the system's needs.
2. It depicts the external view of the system.
3. It recognizes the internal as well as external factors that influence the system.
4. It represents the interaction between the actors.

# ACTIVITY DIAGRAM

An Activity Diagram is a behavioural diagram. It depicts the behaviour of a system. Its primary use is to depict the dynamic aspects of a system. The dynamic aspect of a system specifies how the system operates to attain its function.

It is basically a flowchart to represent the flow from one activity to another activity. Activity Diagrams are not exactly flowcharts as they have some additional capabilities including branching, parallel flow, etc.



# SEQUENCE DIAGRAM

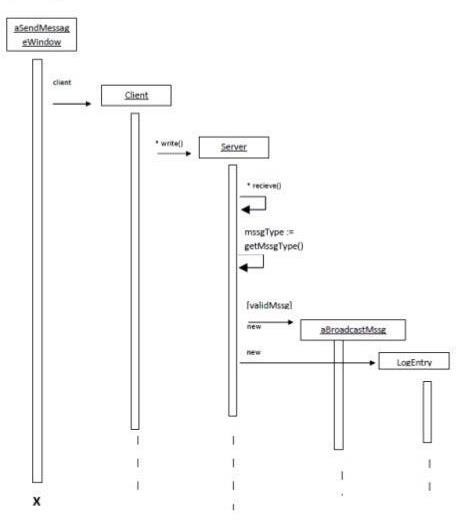


Fig 3.5 Sequence Diagram for Firebase backend

The design shows the detailed illustration of events sequenced and happens in Chatting System. This designed sequence diagram can show programmers and readers the sequence of messages between the actor and the objects.

As you can see through the illustration, the conditions and interactions are emphasized. These interactions are essential for the Online Chatting in System development.

The series of messages are shown and labeled to guide you in building the System. You can modify the design if you have more ideas. You can also add more features to this design and use it as your project blueprint.

* 1. **COLLABORATION DIAGRAM**

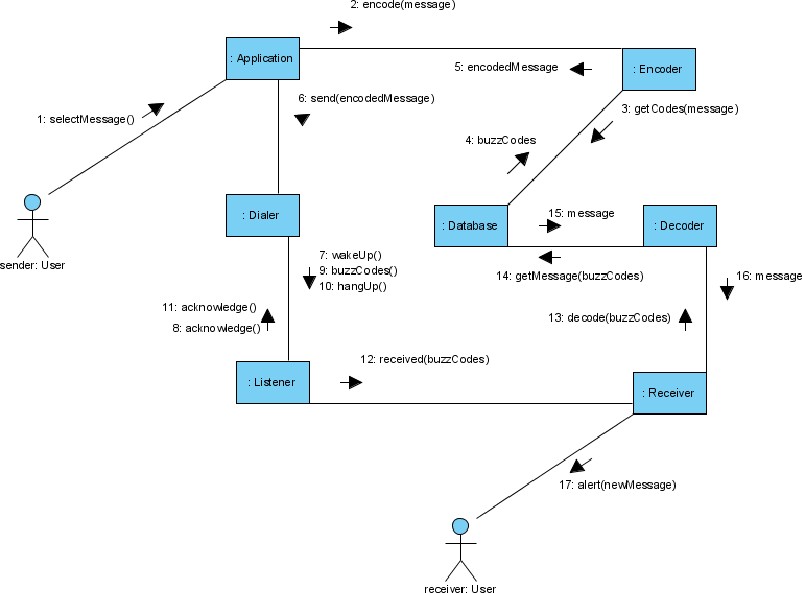


Fig 3.6 Collaboration Diagram for Firebase backend

The collaboration diagram is used to show the relationship between the objects in a system. Both the sequence and the collaboration diagrams represent the same information but differently. Instead of showing the flow of messages, it depicts the architecture of the object residing in the system as it is based on object-oriented programming. An object consists of several features. Multiple objects present in the system are connected to each other. The collaboration diagram, which is also known as a communication diagram, is used to portray the object's architecture in the system.

# STATE CHART DIAGRAM

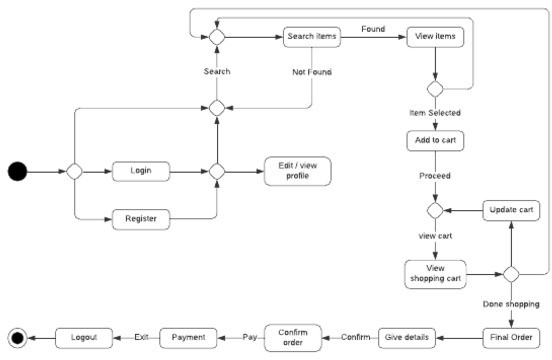


Fig 3.7. State Chart for Firebase backend

The name of the diagram itself clarifies the purpose of the diagram and other details. It describes different states of a component in a system. The states are specific to a component/object of a system.

A State chart diagram describes a state machine. State machine can be defined as a machine which defines different states of an object and these states are controlled by external or internal events.

State chart diagram is one of the five UML diagrams used to model the dynamic nature of a system. They define different states of an object during its lifetime and these states are change.

# COMPONENT DIAGRAM

A component diagram is used to break down a large object-oriented system into the smaller components, to make them more manageable. It models the physical view of a system such as executables, files, libraries, etc. that resides within the node.

It visualizes the relationships as well as the organization between the components present in the system. It helps in forming an executable system. A component is a single unit of the system, which is replaceable and executable. The implementation details of a component are hidden, and it necessitates an interface to execute a function. It is like a black box whose behaviour is explained by the provided and required interfaces.

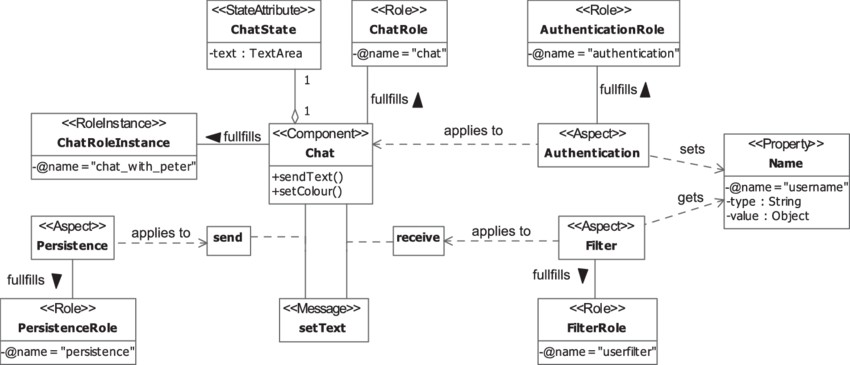


Fig 3.8. USE CASE DIAGRAM for Firebase backend

# DEPLOYMENT DIAGRAM

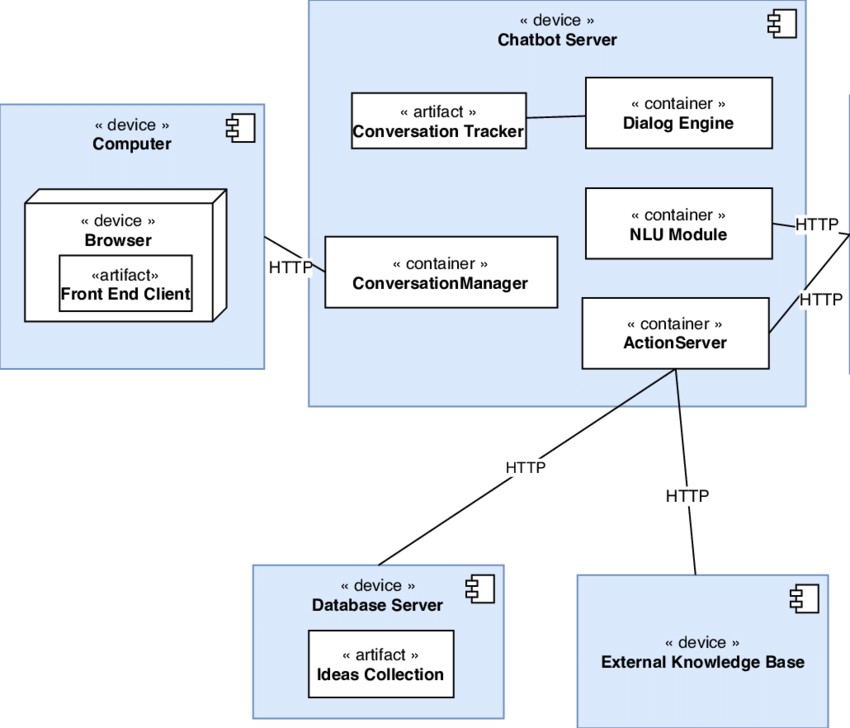


Fig 3.9. Dipolyment Diagram for Firebase backend

Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed.

Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.The term Deployment itself describes the purpose of the diagram. Deployment diagrams are used for describing the hardware components, where software components are deployed. Component diagrams and deployment diagrams are closely related.

# Chapter- 6 CONCLUSION

In conclusion, the Document Scanner project has successfully addressed the objectives outlined in the project report. The utilization of OCR, OpenCV, and PyTesseract has resulted in a versatile and efficient system for document digitization. As technology continues to evolve, this project lays the foundation for ongoing research and development, fostering innovation in the realm of document processing and management.

The Document Scanner project stands as a testament to the synergy between computer vision, OCR technologies, and user-centric design, offering a practical solution for organizations and individuals seeking to streamline their document workflows in an increasingly digital world.

# Chapter-7

### FUTURE SCOPE OF PROJECT

### 1. Enhanced OCR Accuracy:

### Research and implement advanced OCR techniques and models to further improve accuracy, especially in handling complex fonts, languages, and diverse document layouts.

### 2. Machine Learning Integration:

### Explore the integration of machine learning algorithms for adaptive image preprocessing and OCR model training. This can lead to a system that becomes more proficient over time as it encounters different document types.

### 3. Intelligent Document Classification:

### Develop capabilities for intelligent document classification to automatically identify and categorize documents based on content, structure, or context.

### 4. Cloud Integration:

### Implement cloud-based storage and processing to enhance scalability and accessibility, allowing users to store and access their digitized documents securely from anywhere.

### 5. Mobile Application Development:

### Extend the project to include the development of a mobile application, enabling users to capture and process documents using their smartphones, making the system more portable and accessible.

### 6. Integration with Existing Document Management Systems:

### Integrate the Document Scanner with existing document management systems, allowing seamless integration into organizational workflows and enhancing overall document handling efficiency.

### 7. Cross-Platform Compatibility:

### Expand compatibility to include additional platforms and operating systems, ensuring a broader user base and accessibility across various devices.

### 8. Real-Time Collaboration:

### Incorporate real-time collaboration features, enabling multiple users to work on the same document simultaneously and facilitating collaborative document editing and review.

### 9. Blockchain Integration for Document Security:

### Explore the integration of blockchain technology for enhanced document security and tamper-proofing, ensuring the integrity and authenticity of digitized documents.

### 10. Natural Language Processing (NLP):

### Integrate NLP techniques to analyze and understand the context of the extracted text, enabling more intelligent and context-aware document processing.

# Chapter-8

**REFERENCES**

1. Smith, John A. "Optical Character Recognition: Principles and Applications." Springer, 2018.
2. Jones, Mary B. "Handwriting Recognition Algorithms: Applications and Techniques." CRC Press, 2017.
3. Zhang, Liang and Cheng, Yuxin. "A Survey of Optical Character Recognition Systems." Journal of Computer Science and Technology, vol. 32, no. 4, 2017
4. Brown, Emily C. "Image Processing for Document Scanning and Text Extraction." Wiley, 2016.
5. Jurafsky, Dan and Martin, James H. "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition." Prentice Hall, 2019.
6. Li, Xiang and Lu, Shijian. "Deep Residual Learning for Handwritten Chinese Character Recognition." IEEE Transactions on Image Processing, vol. 28, no. 2, 2019, pp. 586-591
7. Rahman, M. A., and Ahmed, M. S. "A Review on Optical Character Recognition (OCR) Techniques." Procedia Computer Science, vol. 112, 2017, pp. 1009-1016.
8. ISO 9001:2015. "Quality Management Systems - Requirements." International Organization for Standardization, 2015.
9. GDPR (General Data Protection Regulation). Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data.