Final Project

Text Extractor: Web Application for Detection and Extraction of text from Images using Machine Learning Techniques

TEAM 6

Tushar Nayan

Srimant Kumar Mohanty

Neeraj Kumar Kondaparthi

Lichi Mahajan

Sanjay Kumar Mucharla

Knight Foundation School of Computing & Information Sciences**,**

Florida International University

CEN5011: Advanced Software Engineering

Dr. Charlyne Walker

December 2023

1. Description of the topic of the project

Let me take you to a scenario in which our grandfathers' generation people had to fill out tax forms or fill out a complex application which required lots of documents references and filling those details in the correct place on the web page to submit the application. With the advancement in responsive web interfaces submitting sensitive data in a faster and more reliable manner is still a challenge and an area with intensive research. We aim to develop intelligent document processing in our applications and optimize the workflows that use applications from Optical Character Recognition (OCR) (Kaundilya et al., 2019). Most of the traditional text extraction algorithms emphasize extracting text from a simple background (monochromatic background or text with distinct colors) (Surana et al., 2022). In this project, we consider training and extracting the text from complicated backgrounds like Passports, State IDs, driving licenses, tax forms (W2, W4, etc.), or other relevant documents to fill the web application forms (for instance hospital/bank appointment/tax forms). We will try in this project to make the web interface smarter and more autonomous. So that an uneducated person or a person who does not know how to use technology can also just click a few snaps of relevant documents on his/her mobile at the fingertip after that our application will process the text and fill out the application on the user's behalf. We also consider the user's privacy for uploading such sensitive documents on a web server. To meet the legal front of the application we will take each user's consent in the form of a digital signature for the document they are uploading. The application could be used in various other forums where text extraction from images can solve novel problems. The application has enormous potential in developing countries where part of digital transformation can help boost the economy, literacy, and digital awareness among the common citizens. The applications will also play a huge role in developed countries

like the USA in industrial service-based uses or RPA (Robotics Process Automation) software and many more.

2. Description of the topic of the project

2.1 Salient characteristics of the customer or sponsoring organization

The project's customer or sponsor will be an entity or group keen on enhancing access to digital services, advancing document processing technology, and protecting data privacy. As a government of rising countries moving towards digital transformation would be interested in streamlining the complex document processing using AI and ML automation. Mostly, government organizations will use this to get huge user capacity to make infrastructure stronger and more reliable. This sort of capable software can push for a lower level of corruption in government service setup. These are some of the salient characteristics.

Whenever a user comes to the third-party website with whom our application has an agreement will show a highlighted area with the option of whether the user wants to use our service or not. If they click on the link, they are then directed to the login page where they can either sign up for a new account or log into an existing one. Once authenticated, the user can proceed to capture a photo or select files, which may include school transfer documents or bank-related materials. Users are prompted to specify the type of documents they are uploading, such as ID cards or certificates, and then, for added security, they grant permission to encrypt data from their device storage or camera. This encryption ensures the safeguarding of their personal information as they upload images containing text, which could be sourced from a smartphone photo, scanned document, or any digital image.

Continuing the journey, our application employs machine learning models to identify text regions within the uploaded images. Subsequently, optical character recognition (OCR) is applied to recognize and extract the text from these identified areas. The user is then allowed to inspect the extracted text that is filled on the website by our application and make any required modifications or revisions to make sure the correct data is filled out. This allows them to verify whether the text is in the right place or not. Once the user is satisfied with the extracted text, they have the option to store this data within our application for future reference unless they choose not to do so. Once they are happy with the data filled in the form, they can sign the digital signature to agree with our terms and conditions and at last, submit the form. Finally, when their task is complete, the user can easily log out of the service or exit the application, concluding their interaction with our user-friendly platform.

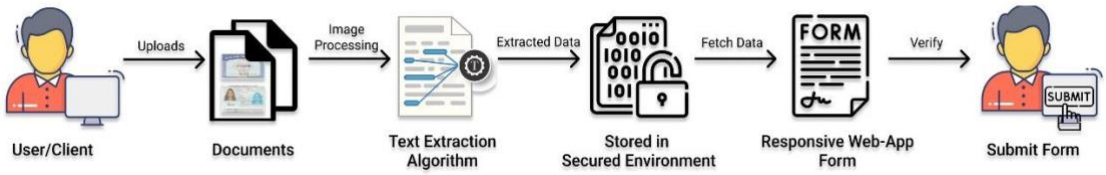

Figure 1: Flow of our project

2.2 Description of the salient characteristics of the application/tool

I. *What is the background of the project idea? What is the problem?*

The background of the project idea is, that in developing countries like India where a considerable number of the population does not know how to operate software tools, not used to handle mobile or laptops. Whereas day-by-day gradated changes are brought by the government of India as a mission towards Digital India. The government

is promoting the use of digital platforms to make processes fast and avoid long in-person queues. Ironically, the common person does not know where to click on the form to get the keyboard pop up, which type of data needs to be filled in which field to satisfy the page constraint ex. a text area needs numbers, not text data, and many more such small and big technical details. Another challenge that stands out is everyone from rural areas does not know how to read and write English, whereas all government-related functions happen in English. This is not a single country's problem. If you pick any non-English speaking country this problem exists there. This big gap can be encountered by our application (Wikipedia contributors, 2023).

Our project tackles the problem of making difficult web interactions and form submissions simpler, especially for people who are less proficient in technology. It was inspired by the challenges experienced by older generations and people who are less accustomed to using sophisticated web interfaces, which frequently call for maintaining many document references and data inputs.

In order to process documents intelligently, especially those with complex backgrounds like passports, IDs, and tax forms, our system makes use of optical character recognition (OCR) technology. This makes it possible for individuals with little technical expertise to complete web application forms effortlessly and independently. Through a digital signature consent procedure for uploaded documents, we prioritize user privacy. Our project acts as a link, empowering people and promoting a more diverse and effective digital environment.

II. *How is your application/tool different from what already exists?*

Our tool is different in many ways. Such a solution to extract text from images is trivial. If we go to the market there are big players like Microsoft, Amazon, Adobe, Tesseract, and many more. They all do the OCR to retrieve text information from the images. Information retrieval is done mostly in 2 common ways, that is, either train the model by showing it lots of similar images and templates, by which the model will remember in which part of the image what information is stored. For example, we will train a model with different images of a driver's license, then the model will slowly start to pick the license number, license holder name, his or her address, etc. by the information position in the image. The second way is to retrieve information in key-pair value. In the second approach, the model crawls through the image text and makes key-pair values based on the relevance and positioning of text in JSON format, which Microsoft is doing right now. The drawback with this type of data modelling is you cannot fill out web forms intelligently.

What is different in our application is we will mix these two approaches to get a sense of data from images and remember the data heuristically like a human brain. So, that it can read all data, then process it, and understand the relevance of the data. After that fill in the required information in the specified asked fields of the forms. Without human intervention, the application is able to do all the boilerplate data fill process by itself.

Another great advantage is to be cost-wise. All the big tech giants' solutions are subscription models. They charge a good chunk of

money, which is difficult for an unprivileged person to afford. We will make it cost-free for all government uses.

III. *What are the implications of the tool you develop?*

Our software tool will have a tremendous impact on society at a global level. If we talk about India, it has a huge population of 1.4 billion people. If 40% of people use it for their application automation process to reduce the hectic effort of manual data entry, then the number goes to 55 million. Considering the world population is 8.5 billion, according to the World Bank's English Proficiency Index (EPI) is just 53.4% in 2022, which is moderate. According to the IMF definition, there are 152 developing countries with a current population of around 6.82 billion. This means that about 83% of the world's population lives in developing countries. All these developing countries would highly benefit from our application. If our application can reach an accuracy of more than 90% and if it can reach out to 30% of the developing population who have less knowledge about technology and do not know English, this software will be eventually used by 2 billion people across the globe. This can address all those overwhelming fears of "Am I filing out the correct data or not?", in a particularly important government subsidy form or tax form.

If we take a case study of developed and English-proficient parts of the world this technology can definitely be used as a service to improve productivity and address future problems where the text to fill data into a form is a concern.

3. **Other important contextual issues especially any external constraints placed on the application/tool.**

Our software tool must carefully consider a complex set of contextual limitations in order to operate properly. The treatment of personal data is a top priority, and different data privacy and security rules apply depending on where and how the system is used. The handling, processing, and storage of personally identifiable data are governed by strict laws like GDPR, HIPAA, or local regulations. Second, the system's performance may be impacted by the quality of images taken in difficult situations, such as glare and low light, which may impede its capacity to accurately process information. Additionally, while observing data protection rules, we have to deal with a number of legal and ethical issues, such as user consent, data retention policies, and potential technological misuse. (Bern et al., 2007) These contextual concerns highlight how crucial it is for us to be committed to both privacy and functionality while developing and distributing our software.

4. **What programming language(s) will be utilized for the project?**

Text extraction from images can play a crucial role in finding vital and valuable information in no time. After an intensive study of current tools and technology available, we decided to use MERN Stack. The MERN stack is a popular technology stack for building web applications. It consists of four main technologies: MongoDB, Express.js, React, and Node.js. Here's a brief overview of each tool and technology we will be using at the development phase.

Development Stage	**Tools**
Machine Learning Model	Optical Character Recognition (OCR), OpenCV, Numpy, Keras-OCR

Frontend	HTML, CSS3, Bootstrap4, DOM
Backend	DotNet, JavaScript, Python
Database	SQL
Other Tools & Framework	Azure, Figma

Table 1: Tools & Technology

5. Outline a proposed schedule for your project

5.1 Show your User Story backlog

5.1.1 The user stories should follow the format: As a ,<type of user>, I want <some goal>, so that <some reason>.

5.1.2 It should also be followed by a set of Acceptance Criteria: Verify that…

User Story Backlog List:

End User's User Stories

1. User Registration

As an unregistered user, I want to be able to register into the application, so that I can use and access the application features and services.

Acceptance Criteria:

- Verify that user registration requires a valid email, Full name, and a password complying with the password acceptance standard
- Verify that the user should receive a confirmation email upon successfully registering into the application
- Verify that the registration form validates all the checks for valid email format, password standard

2. User Login

As an existing user, I should be able to log into my account, so that I can access all my information

Acceptance Criteria:

- Verify that the login process requires a valid email address and password.
- Verify that users can reset their password if they forget it.
- Verify that the login form includes validation checks for valid email format and password entry.
- Verify that proper access to the logged-in user is provided

3. User Captures or Upload Document

As a user, I want to be able to capture images or upload documents containing relevant information, so that I can extract and use the text data.

Acceptance Criteria:

- Verify that the application allows users to capture images using the device's camera or upload documents from their local machine.
- Verify that the captured/uploaded images/documents are processed for text extraction.

4. ML Model Extracts Relevant Information from Uploaded Images

As a user, I want the ML model to extract only the relevant information from the uploaded images so that the data collection process is efficient and accurate.

Acceptance Criteria:

- Verify that the ML model accurately identifies and extracts relevant information from the images.

- Verify that the model filters out irrelevant text and focuses on key information.

5. Application Fills Web Page Fields with Extracted Information

As a user, I want the application to intelligently fill third-party website forms with the extracted information, so that I can complete tasks seamlessly.

Acceptance Criteria

- Verify that the app communicates with third-party websites through API to retrieve form fields that the app needs to fill out for user flexibility
- Verify that the app intelligently maps and fills the required fields with the extracted information.

6. Users Can Verify Filled Fields

As a user, I want to be able to review and verify that all fields on the web page are correctly filled with the extracted information before submission.

Acceptance Criteria

- Verify that the application displays the filled form fields in a clear and organized manner for user review.
- Verify that the user can easily navigate through the filled form fields.

7. Users Can Correct Incorrect Information

As a user, I want the ability to manually correct any inaccuracies in the filled information, so that it is accurate and complete.

Acceptance Criteria:

- Verify that the user can edit individual pieces of extracted or filled information.
- Verify that the corrected information is updated and reflected in real-time.

8. Users Save Data into the Database

As a user, If I give consent, I want to securely store my data in the database using encryption, tokenization, and masking so that it is protected from unauthorized access.

Acceptance Criteria:

- Verify that data stored in the database is encrypted, tokenized, and masked to ensure security.
- Verify that sensitive information is not exposed in its raw form.
- Verify that access to the database is restricted to authorized personnel only.

9. Users Can See Their Stored Personal Information

As a user, I want to be able to view all my personal information stored in the application, so that I can review and verify its accuracy.

Acceptance Criteria:

- Verify that the application provides a user-friendly interface for viewing stored personal information.
- Verify that the displayed information is accurate and up to date.

10. Users Can Update Their Information

As a user, I want the ability to manually update my personal information stored in the database, so that it is accurate and reflects any changes in the future.

Acceptance Criteria:

- Verify that users can edit individual pieces of their personal information.
- Verify that the updated information is reflected in real-time.

11. Users can Sign a Digital Signature for Data Consent

As a user, I want the ability to provide a digital signature as consent for the use of my data and store it in the cloud for future use.

Acceptance Criteria:

- Verify that the application provides a secure method for users to create and submit a digital signature.
- Verify that the digital signature is securely associated with the user's consent for data use.

12. Users Can Submit the Form

As a user, I want to be able to submit the filled form after verifying the information and providing consent.

Acceptance Criteria:

- Verify that the application provides a clear and intuitive "Submit" button for users to initiate the submission process.
- Verify that upon submission, the form data is securely transmitted to the third-party website.

Admin Point of User Stories

13. Admin Can Manage User Accounts

As an admin, I want to be able to manage user accounts, including creating, suspending, or deleting them, to maintain a secure and compliant platform.

Acceptance Criteria:

- Verify that the admin has access to a user management dashboard.
- Verify that the admin can create new user accounts with a valid email address and password.
- Verify that the admin can suspend or reinstate user accounts as needed.
- Verify that the admin can delete user accounts if required.

14. Admin Can Manage Third-Party and User Agreements

As an admin, I want to be able to manage agreements between the application and third-party services, as well as user agreements, to ensure legal compliance and protect user rights.

Acceptance Criteria:

- Verify that the admin has access to an agreement management interface.
- Verify that the admin can view, update, or create new agreements with third-party services.

15. Admin Can Support in Case of Any User Data Issue

As an admin, I want to provide support in case of any user data-related issues, to ensure data integrity and user satisfaction.

Acceptance Criteria:

- Verify that the admin has access to a user data support interface.
- Verify that the admin can view reported user data issues and their details.
- Verify that the admin can take appropriate actions to resolve the reported data issues.

16. Admin Can Support in Case of Any Software Misbehaves

As an admin, I want to be able to provide support in case the application misbehaves or does not function as expected, to ensure uninterrupted service for users.

Acceptance Criteria:

- Verify that the admin has access to a software support interface.
- Verify that the admin can view reported software issues and their details.
- Verify that the admin can take appropriate actions to address and resolve reported software misbehaviors.

17. Admin Can Handle Customer Support Requests

As an admin, I want to be able to view and respond to customer support requests, so that I can provide timely assistance to users.

Acceptance Criteria:

- Verify that the admin has access to a customer support dashboard.
- Verify that the admin can view customer support requests, including their details and urgency.
- Verify that the admin can respond to support requests or escalate them if needed.

18. Admin Can Handle Data Privacy and Storage

As an admin, I want to ensure secure data storage in the cloud, with proper access controls, to protect user information and comply with data privacy regulations.

Acceptance Criteria:

- Verify that the admin has access to a data privacy and storage management dashboard.

- Verify that the admin can configure access controls for cloud storage, restricting it to authorized personnel only.
- Verify that the admin can monitor and ensure compliance with data privacy regulations.

Third-Party Website User Stories

19. Third-party websites Offer Access to Our Application

As a Third-Party Agent, I want the selected third-party websites to provide an option for their users to access our application directly from their platform, should they choose to use our services.

Acceptance Criteria:

- Verify that the Indian third-party websites incorporate a seamless option, such as a link or button, that directs their users to our application for additional services or data processing if desired.

20. Integration with Three Third-Party Websites

As a Third-Party Agent, I want our application to integrate seamlessly with three chosen Indian third-party websites, enabling users to efficiently fill in the required details on these websites.

Acceptance Criteria:

- Verify that our application successfully establishes API integration with the selected Indian third-party websites.
- Verify that the API provided by the third-party websites allows access to all necessary fields for data submission.

21. API Access Provided by Third-Party Websites

As a Third-Party Agent, I want the selected third-party websites to provide API access for all fields on their pages that need to be filled, ensuring smooth interaction with our application.

Acceptance Criteria:

- Verify that the third-party websites provide comprehensive API documentation, including details of all required fields and endpoints for data submission.
- Verify the API input requirements and out success code.

System Level Functionality User Stories

22. Accurate Extraction of Important Data from Uploaded Images

As a user, when I upload multiple images, I expect the application to accurately extract all important data from them.

Acceptance Criteria:

- Verify that the application successfully extracts relevant data from the uploaded images.
- Verify that the extracted information is accurate and complete.

23. Fill in Extracted Information in the Required Web Page Fields

As a user, after the extraction of information from uploaded images, I want the application to intelligently fill the required fields on the web page with this extracted information.

Acceptance Criteria:

- Verify that the application communicates with the web page to retrieve the required fields.
- Verify that the application accurately maps and fills the necessary fields with the extracted information.

24. Secure Storage of User Data in Azure Cloud Storage

As a developer, I want to ensure that user data is stored securely in Azure cloud storage so that it is protected from unauthorized access or tampering.

Acceptance Criteria:

- Verify that user data is encrypted both in transit and at rest using industry-standard encryption protocols.
- Verify that access to the Azure cloud storage is restricted to authorized personnel only and is protected using robust access controls and authentication mechanisms.
- Ensure that user data is regularly backed up to prevent data loss in the event of a system failure or security incident.
- Implement logging and monitoring to track access to the data, including any unauthorized attempts or suspicious activities.

5.2 Describe the process used to determine user story allocation

User story allocation is a key step for our project development and management. It is done effectively, which ensures the time spent on our work results in quality delivery. It began with the making up of the product backlog which includes the full and organized list of the user stories and tasks which need to be implemented. We have assessed the team's ability and availability to spend the time that decides the number of user stories that need to be included in the sprint. Then comes the story points estimation, in which the story points are allocated to each user story in the sprint which conveys the effort needed by the team to implement it. Each user story can be broken down further into tasks at the developer's convenience. The same process can be repeated for the next sprints.

Story points	Story
1	• Users can verify whether all fields are filled with correct data. • Users can submit the form
3	• Users can see their personal information stored in the application.

	• Admin can handle all data privacy and data storage in the cloud with proper access. • The user captures or uploads documents.
5	• User Registration • User Login • Users save data into the Database. Data will go into the database using encryption, tokenization, and masking. • Users can update their information. • If information is not filled in correctly then have the leverage to correct it • User should be able to sign the digital signature for consent of data usage. • Admin can manage user accounts. • Admin can manage third-party and user agreements. • Admin can support in case of any software misbehaves it doesn't work as expected. • Admin can handle customer support requests. • Third Party customers will provide the API for all the fields on the page that need to be filled. • Third party customers will provide access to our application for filling in details.
8	• As a developer, I want to ensure that user data is stored securely in Azure cloud storage so that it is protected from unauthorized access or tampering. • When the user uploads a few images, it should be able to extract all important data correctly. • Fill the extracted information in the required field of the web page. • Admin can support in case of any user data issue.
13	• When the user uploads the images the ML model will render through the image and collect all the important information in key pair value • Then the application will fill the web page with appropriate information in the required field

5.3 Determine the sprint period (1 – 3 weeks)

We have determined the sprint period of 2 weeks based on project complexity, team member experience, and team member availability to work on the project.

Please find below the sprint backlog with the respective story. If in case for any reason the story implementation is not achieved in that week, we have the flexibility of pushing it to the next coming sprint to implement it. Please note that the basic testing of the implemented story happens immediately.

5.4 Allocate the backlog to the Sprints

SPRINT 1 (Week 1 and Week 2)	Week	Developer
• User Registration	1	Neeraj
• User Login	1	Sanjay
• The user captures or uploads documents.	1	Neeraj
• Admin can manage user Accounts. Admin can manage third-party and user agreements.	1 2	Tushar Sanjay
• To start with we will be planning to take 3 third-party Indian websites.	2	Neeraj
• Third party websites will provide the API for all the fields on the page that need to be filled. Third party websites will provide access to our application for filling in details	2 2	Lichi Lichi

SPRINT 2 (Week 3 and Week 4)	Week	Developer

• When the user uploads the images the ML model will render through the image and collect all the important information in key pair value • The application will fill the web page with appropriate information in the required field. • Admin can support in case of any software misbehaves it doesn't work as expected	3 4 4	Tushar Tushar Srimant

SPRINT 3 (Week 5 and Week 6)	Week	Developer
• Users save data into the Database. Data will go into the database using encryption, tokenization, and masking. • Users can see their personal information stored in the application. Users can update their information. Admin can support in case of any user data issue. • When the user uploads a few images, it should be able to extract all important data correctly. • Fill the extracted information in the required field of the web page	5 5 5 6 6 6	Lichi Tushar Sanjay Srimant Tushar Srimant

SPRINT 4 (Week 7 and Week 8)	Week	Developer

• Users can verify whether all fields are filled with correct data. • If information is not filled in correctly then have the leverage to correct it • User should be able to sign the digital signature for the consent of data use. • • Users can submit the form. • Admin can handle customer support requests. • Admin can handle all data privacy and data storage in the cloud with proper access. • As a developer, I want to ensure that user data is stored securely in Azure cloud storage so that it is protected from unauthorized access or tampering	7 7 7 8 8 8	Neeraj Sanjay Lichi Sanjay Srimant Srimant Lichi

5.5 Show your burndown chart with your planned sprints for the entire semester.

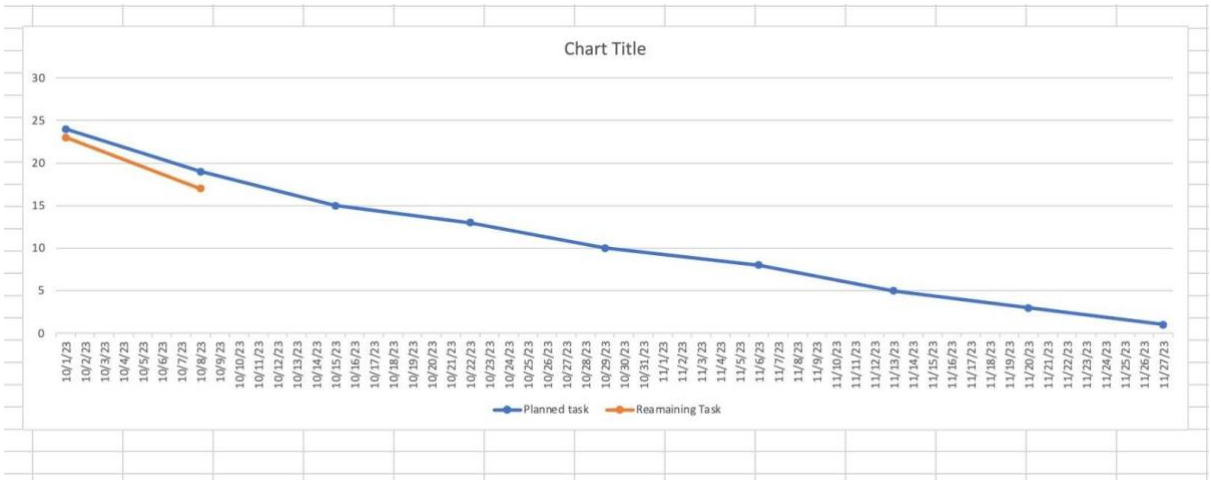

Figure 2: Burndown Chart

6. Describe your team and the proposed roles and responsibilities for your teammates

Our team members are constantly researching the relevant information essential and have a well-rounded mix of skills and expertise, which is a crucial aspect for the success of the project.

Our team have a total of five members and will be working closely on different aspects of development in the following way –

Member Name:	Tushar Nayan
Role:	Project Manager and Machine Learning Expert
Responsibilities:	Overall project management and coordination. Developing and training machine learning algorithms. Guiding and mentoring team members in machine learning tasks. Ensuring project milestones and deadlines are met. Collaborating with Lichi (Team Member) on dataset collection and processing.

Member Name:	Lichi Mahajan
Role:	Dataset Manager and Documentation Specialist
Responsibilities:	Collect, curate, and manage datasets for training and testing the machine learning models. Data preprocessing, including annotation and labeling. Collaborating with Tushar on data preprocessing and feature engineering. Documenting project progress, processes, and findings. Keeping records of datasets used and their sources.

Member Name:	Srimant Kumar Mohanty
Role:	Backend Developer and Coordinator
Responsibilities:	Developing the backend infrastructure of the project using Node, Express, Javascript, etc. Handling server-side logic and database management. Ensuring the backend is robust, scalable, and performs well. Collaborating with the front-end team and machine learning expert to integrate their requirements. Coordinate with other team members to resolve their issues.

Member Name:	Neeraj Kumar Kondaparthi
Role:	Frontend Developer and UI/UX Designer
Responsibilities:	Frontend Developer and UI/UX Designer Designing and developing the user interface (UI) of the application. Creating responsive and visually appealing web pages. Working closely with the backend team to integrate frontend and backend components. Collaborating with Sanjay for seamless integration of UI and backend.

Member Name:	Sanjay Kumar Mucharla
Role:	Frontend Developer and UI/UX Designer
Responsibilities:	Collaborating with Neeraj on UI/UX design and development. Implementing interactive features and user-friendly interfaces. Ensuring the application is user-friendly and visually appealing.

PART 2: DESIGN

Definition of Design:

Design is a multifaceted and iterative process that encompasses the conception, planning, and execution of creative solutions to address specific problems or fulfill defined objectives. It involves a deliberate consideration of form, function, aesthetics, and user experience, with the aim of creating products, systems, or experiences that resonate with intended audiences and stakeholders (Cooper, Reimann, & Cronin, 2007).

Our Design Process Journal:

For our project, we mostly followed the structured and process-driven design process. Below we have explained each step in the design process in more detail with the significance of that process and how it helped us decisions for the design process. We also explained the challenges and learning from each step.

i. Project Scope and Definition:

We have Started by outlining the objectives and scope of the project. Clearly state the goals that the online application is meant to accomplish. In this instance, we are trying to make users' lives easy by leveraging the power of machine-learning techniques to extract text from photos and use that inside a user-friendly web application to make it available for users. The scope of the project would be to be able to override third-party websites with our tool to fill the text extracted into the application for user benefit.

ii. Identifying Requirements:

Gather the detailed requirements in the view of end users, stakeholders, and admin prospects. This entails being aware of the kinds of photos that users will submit and the format in which the output is anticipated. It will help us snap a deal and agreement with third-party websites for the override of use. We will be able to finalize the terms and conditions to be agreed between both parties.

This is the phase where we discuss with the stakeholders the product features, application behaviors, and all other important aspects like duration, cost, and human resources.

The requirement gathering in software design is always divided into 4 stages Elicitation, Analysis, Documentation, and Validation.

iii. Data Collection and ML model selection:

To train the machine learning model, compile and prepare a dataset of pictures with required text information. The quality of training data must be guaranteed, and this requires data preprocessing. Select the proper deep learning or machine learning models for text extraction and detection. Factors like accuracy and speed are taken into consideration.

For that, we have decided to use the OCR model for text extraction, where the data set is available in Kaggle. For text detection and pairing, we will use the Microsoft Transformer model and Microsoft SDK. This ML model and Service matched our requirements.

iv. Web Application Architecture:

At this point, we have determined the web development technology stack which comprises databases, front-end and back-end frameworks, and any other tools required for the web application.

We will use the MERN (MongoDB, Express.js, React, Node) JavaScript framework. For DB hosting we will use Microsoft Azure. We will implement it using Microservice Architecture. Where we will modularize each service. We have decided to use the SOLID principles. We want to make our project and code base easy to maintain going forward. We will take care of authentication using OpenID Connect and JWT.

Modularization, Abstraction, Encapsulation, and Separation of Interface and implementation of all is what we have designed. We also focused on using the C4 model in our development process.

v. UI Design and Development:

A user-friendly interface for users to upload images and view extracted text. We must make sure that the design is responsive for both mobile and desktop users. Develop web application, including the logic for processing and forwarding images to the machine learning model. User authentication and authorization features must be implemented. React and Redux will handle the UI performance.

vi. Text Detection and Extraction:

This is the crucial step of the design process where we must integrate the trained machine learning model into the web application. Implement the text detection and text

extraction algorithms to locate and convert text regions from images. This is the core feature of the application.

vii. Testing and Quality Assurance:

Conduct thorough testing, including unit testing, integration testing, and user acceptance testing. Verify the accuracy and performance of text extraction. ML models need to optimize for performance if necessary.

We will conduct the integration testing using Docker and Jasmine.

viii. Deployment and Hosting:

Deploy the web application on a scalable infrastructure, such as cloud services, to ensure it can handle a growing user base and large image loads.

We will use the Microsoft Azure as our IAAS. Where we can store our data furthermore, we will set up the CI/CD pipeline and conduct smooth deployment.

ix. Security and Privacy:

To safeguard user information and photos, security procedures must be put in place. Also addressed issues with data access and storage privacy. There is no compromise on our data privacy as we are taking in sensitive information.

Data privacy and user access to the proper set of data are our focus. We designed that phase with utmost care where scripting and masking are considered in development.

To produce a reliable and user-friendly text extraction web application, we as web developers, machine learning engineers, UI/UX designers, and domain specialists have collaborated during the "Text Extractor" project's design phase.

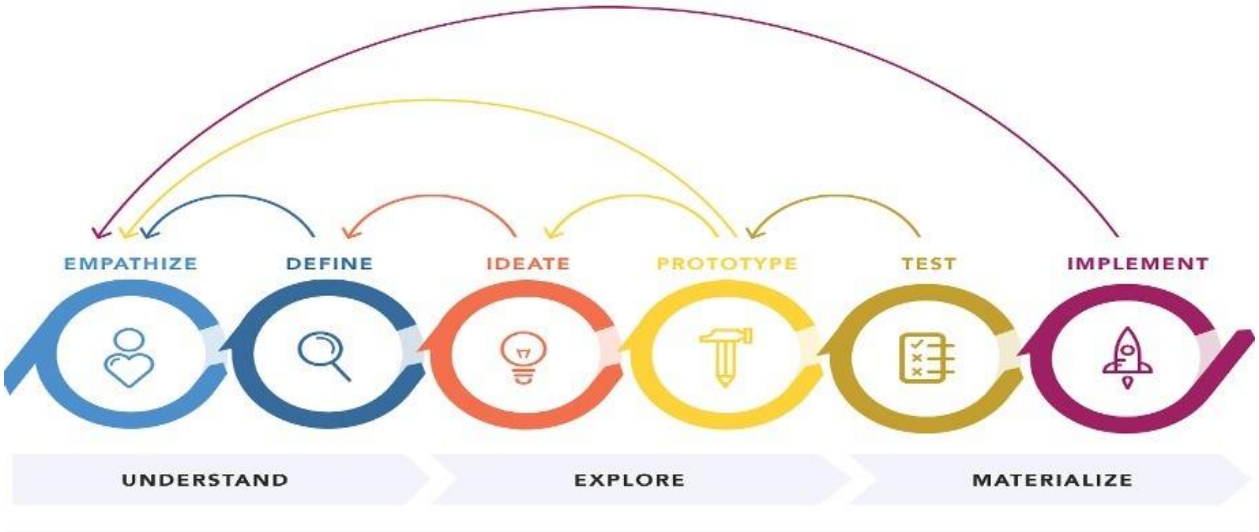

Figure 3: Design Flow Processes

Rational Management:

In the development process of our web application, "Text Extractor," which is designed to identify and extract text from images using machine learning techniques, several strategic choices were made to address critical issues and enhance its overall performance. The primary concerns revolved around ensuring the precision and dependability of text extraction, the ability to scale up to meet growing user demands, and providing a user-friendly experience. After assessing various options, we decided on a hybrid strategy, combining Tesseract OCR for text recognition with a custom Convolutional Neural Network (CNN) for text detection, this CNN-based model is a Microsoft-backed model. This balanced approach strikes a chord between accuracy and performance.

We chose a cloud-based architecture to address scalability since it allows us to dynamically shift resources as user needs vary. Features like drag-and-drop functionality are part of the user interface that was designed to be straightforward to use. These decisions were based on considerations such as accuracy, scalability, and user experience. Within the development team, there were debates regarding choices like custom models versus existing OCR libraries, scalability options, and UI design. These discussions ultimately led

to the development of a robust text extraction application that puts users at the center of the experience.

On cloud service, we initially thought AWS a great option but with time we realized it comes with a cost and it needs meticulous server load balance to meet the free tier. Keeping that in mind we switched to AWS which gives free services to students. Another factor that forced this change in decision is the lack of skill among the team members. We were not confident about the use of AWS. In Microsoft Azure, we have a level of exposure.

While deciding the Database schema we first thought we would use manual scripting and generation of database tables. But we then decided we would use ORM (Object Relational Mapper) in the backend to maintain a smooth DB handle and lightweight framework with low code functionality. We had a lot of debate on what should be our DB schema. What tables do we want to use, what relationship among objects we will have, how we will handle complex dependency among tables, and many more such discussions we had.

Regarding UI design we always kept the principle it should be more visual, and interaction driven. As lots of users are new to digital platforms it should not throw a lot of text content, as they will find it tedious and unnecessary to digest. It should be appealing to the eye to use and at the same breadth, it should be clean and simple to use. At this phase, we had some challenges finalizing a wireframe and a few tussles on personal preference for UI design and look and feel. But we were able to convince ourselves to come to a single thought.

The security of data and its protection has always been a big issue and challenge. Although we are using Microsoft Azure as our cloud platform, so that gives us the first layer of protection from attackers. For added level protection, we had a few debates on using which scripting technique and tokenization framework to script sensitive data into DB.

Throughout the discussion, we always focused on using frameworks and technologies that will make future scalability and maintainability easier and more comprehensive.

The UML Diagrams:

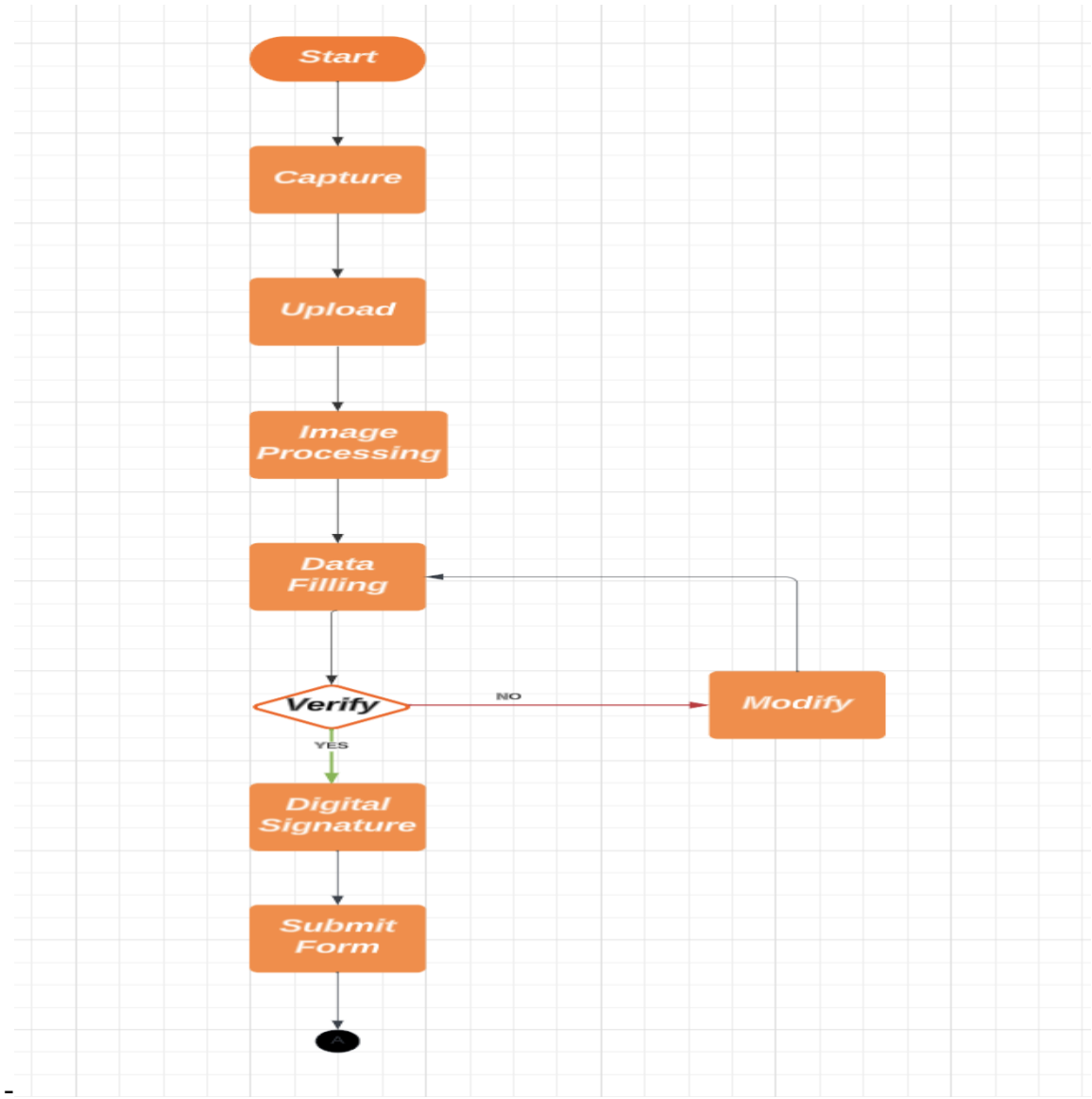

Figure 4: Activity Diagram

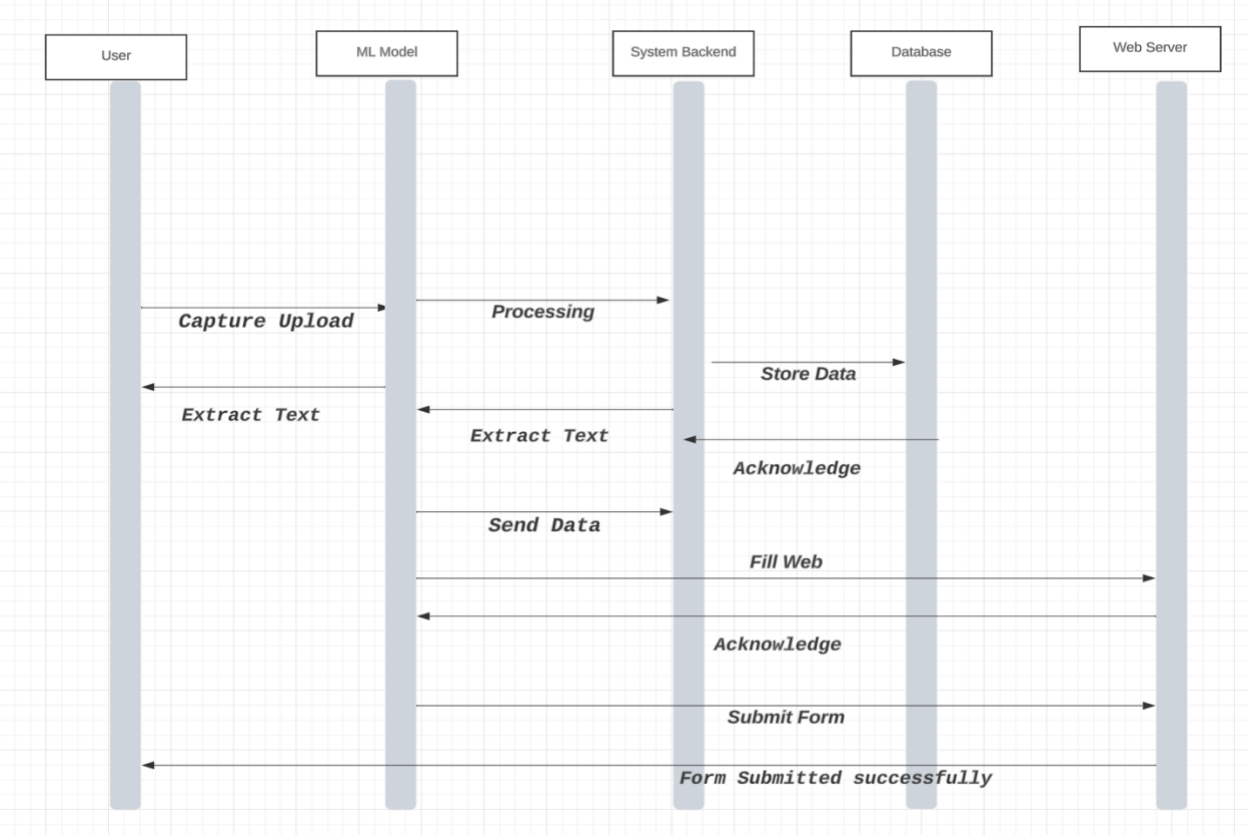

Figure 5: Sequence Diagram

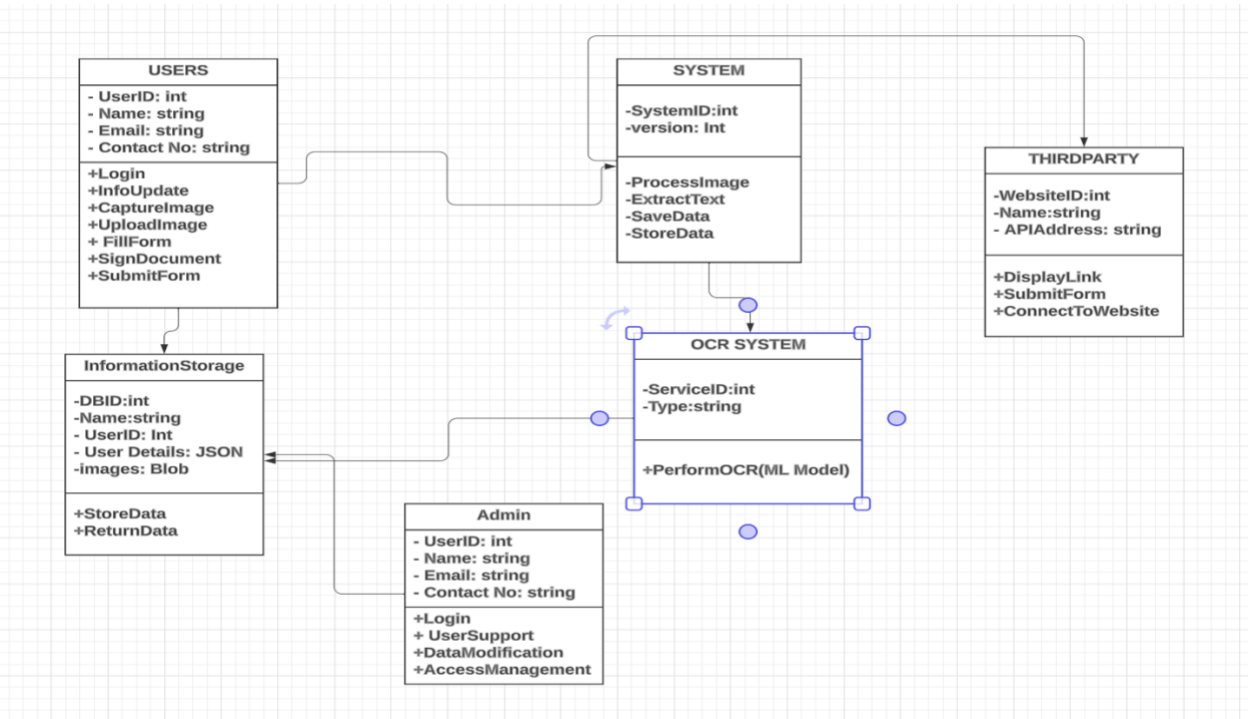

Figure 6: Class Diagram

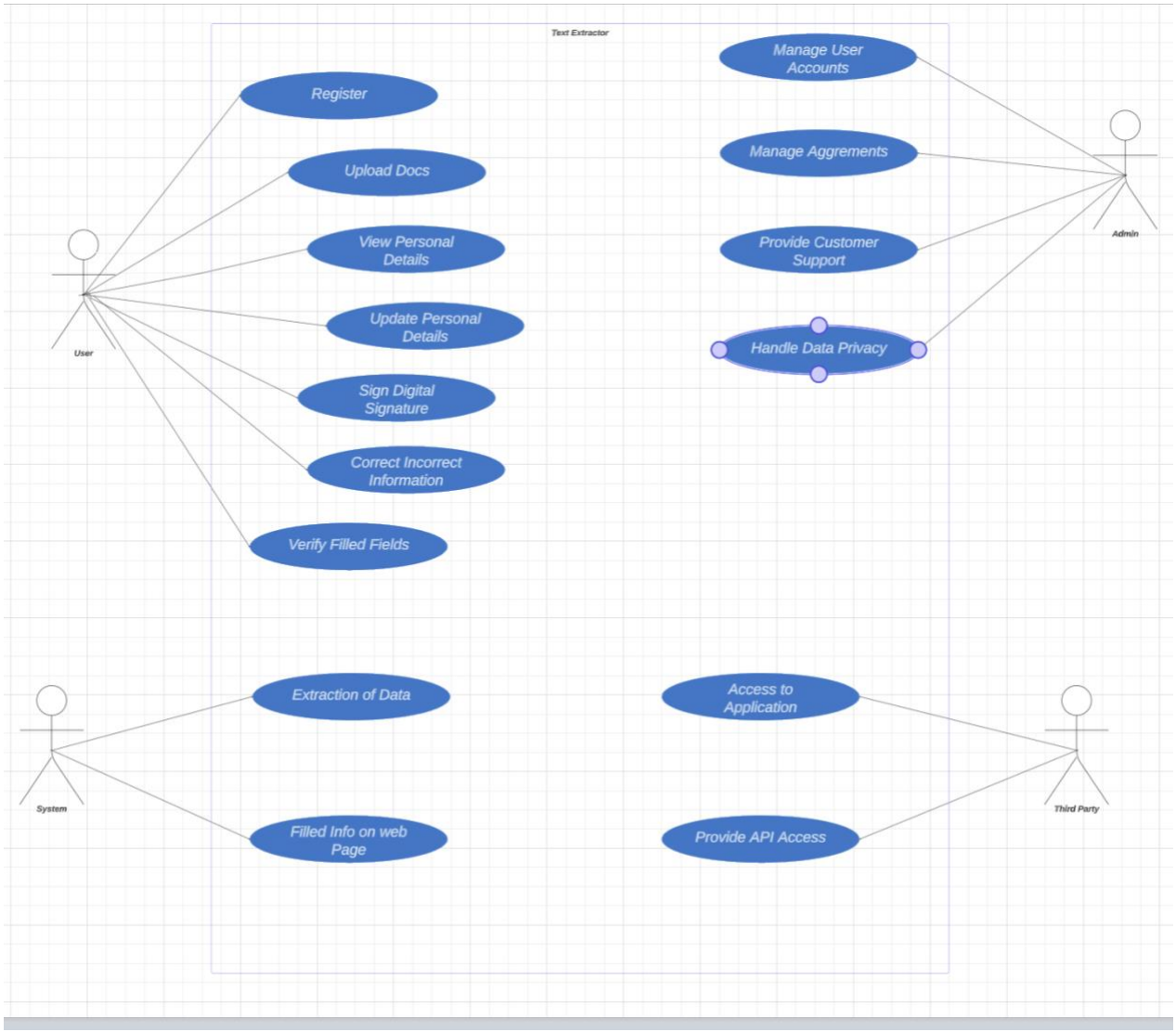

Figure 7: Use Case Diagram

PART 3: VERIFICATION

1. Verification:

Verification is a crucial aspect of development, it acts as a criterion to measure the performance result of the software. “Boris Beizer in his study highlights the significance of verification in the testing procedure”. Reviews and inspections are examples of verification activities that are crucial for finding and fixing flaws early in the development lifecycle. Verification is a crucial step in our research as it guarantees the correct extraction of text from images by the machine learning models and the proper operation of the web application (Beizer, 1990).

Foundations of Empirical Software Engineering: The Legacy of Victor R. Basili:

The book "Foundations of Empirical Software Engineering" highlights the importance of verification in establishing software reliability in the context of empirical software engineering. By verifying the accuracy and resilience of the text extraction system, the project's verification procedure enhances its dependability (Basili et al., 2005).

In conclusion, the project's verification process entails methodical assessments to verify that the created system satisfies predetermined criteria, guaranteeing precise text extraction, safe data processing, and general system dependability. The importance of verification in producing high-quality software systems is supported by academic research.

2. Introduce your verification process – the process invoked by your team

Our verification procedure is a thorough and methodical method designed to guarantee the text extraction system's correctness, dependability, and general quality. Before proceeding to further phases of development or deployment, the procedure is meant to verify that the system satisfies its stated requirements and performs as planned.

1. *Requirement Verification:*
    - Start by carefully going over and verifying that the system requirements, as stated in the user stories and project scope, are precise and well-understood.
    - Review requirements with stakeholders, including end users, to make sure expectations are in line and agreed upon.
2. *Verify the system's design:*
    - This includes the database schema, user interface, integration of the machine learning model, and overall system architecture.
    - Make that the design complies with usability guidelines, best practices, and the intended user experience.
3. *Code Review and Static Analysis:*
    - Conduct code reviews to evaluate the implemented code's quality, readability, and conformity to coding standards.
    - Use static code analysis to find possible problems, like security flaws or code smells.
4. *Unit Testing*:
    - To ensure that every component of the system works as intended, developers create unit tests for each one.
    - Throughout the development process, automated unit tests are run frequently to identify and fix problems early.
5. *Integration Testing:*
    - Verifying the interactions between the machine learning model and the web application was a critical area of focus. Testing the data flow and communication protocols between various elements was part of this.

- Integration tests encompass situations in which different services or modules interact, such as the web application's and the machine learning model's interaction.

6. *System Testing:*
    - Test the system from beginning to end to ensure that it satisfies user needs and functions as planned in an actual setting.
    - Evaluate a range of use scenarios, such as text extraction from various picture formats and communication with external websites.
7. *User Acceptance Testing (UAT):*
    - Conduct UAT with end users to get their opinions on the usability, functionality, and general user experience of the system.
    - Utilize UAT input to inform necessary modifications and system enhancements.
8. *Security and Privacy Verification:*
    - Confirm that user information is adequately protected by the use of security measures such data encryption, access controls, and safe data storage.
    - Make sure privacy laws are followed, and get the permissions required for processing data.
9. *Documentation Verification:*
    - Verify the accuracy and timeliness of all project documentation, including user manuals, system architecture documentation, and code documentation.
10. *Deployment Verification:*
    - Conduct last-minute checks prior to deployment to make sure everything goes according to plan and the system functions properly in the production setting.

Our team hopes to satisfy end users and stakeholders by providing a dependable and high-quality text extraction system by adhering to this verification method. Teamwork, frequent testing, and a dedication to ongoing improvement based on input and knowledge acquired during the development lifecycle are all part of the process.

3. **Describe/demonstrate your test cases – system-level test cases ONLY (no unit testing necessary)**

    1. **User Authentication Test Cases:**

        *Positive Authentication***:** Verify that a user can log in with valid credentials.

        *Negative Authentication***:** Attempt to log in with invalid credentials and ensure access is denied.

    2. **MODEL Test Steps:**

        Data Selection: Choose a representative set of images from the dataset that specifically vary in lighting conditions.

        Text Extraction: Use the model to extract text from each selected image.

        Accuracy Assessment: Compare the extracted text against the ground truth for each image. This involves checking for correctness in character recognition, key-value pair, and overall textual context.

    3. **Backend Data Flow from Database to User Interface Test Cases:**

        Data Retrieval Accuracy: Confirm that the data displayed on the UI matches the data stored in the database.

        Inter-Service Communication: Test the communication pathways between microservices for data consistency and error handling.

        API Gateway Functionality: Confirm that the API gateway correctly routes requests to appropriate services and handles responses.

        Error Handling: Test how the system handles database retrieval errors (e.g., connection failures, data inconsistencies).

Performance: Evaluate the system's response time when retrieving large data.

4. **Data Flow from User Interface to Database Test Cases:**

   Data Upload: Verify that new data entered through the UI is correctly stored in the database.

   Data Update: Ensure that modifications made through the UI are accurately reflected in the database.

5. **Data Fill Accuracy Test Cases:**

   Field Validation: Check that all required fields are correctly validated for format and type before submission.

   Default Values: Confirm that default values are correctly populated where applicable.

   Boundary Conditions: Test the system’s handling of edge cases, like extensive inputs, wrong format, etc.

6. **Data Extraction Accuracy Test Cases:**

   Extraction Quality: Assess the accuracy of text extraction from the AADHAR CARD image using ML.

   Format Handling: Test the system's ability to handle and extract text from different image formats (JPEG, PNG, etc.).

   Error Reporting: Ensure the system reports errors or uncertainties in text extraction correctly.

7. **Third-Party Integration Test Cases:**

   Integration Functionality: Confirm that all integrated third-party services (bank website, new sim acquire, etc.) work as expected with our application.

   Data Sync: Test the synchronization of data between your system and third-party services.

Error Handling: Evaluate how the system handles failures or disruptions in third-party services.

8. Data Storage Security Test Cases:

Access Control: Verify that only authorized users can access READ stored data.

- *ADMIN:* Admin can access read and modify stored data
- *USER:* The user can access read data but cannot modify but the user can modify data through the application.

Encryption: Test if sensitive data is appropriately encrypted at rest and in transit.

Compliance: Ensure the data storage complies with relevant data protection regulations.

In summary, this comprises tests for service isolation, inter-service communication, and system resilience to validate the microservices architecture. Request handling, business logic correctness, data consistency, and efficient database interface are all examined at the Controller, Service, and Model levels. Test cases for machine learning models, especially OCR (Optical Character Recognition), are meant to evaluate data preparation, model training, accuracy, generalization, and performance under a variety of scenarios, such as varying illumination in photographs. This thorough testing technique assures not just the operation and dependability of individual components, but also the system's smooth integration and overall performance.

The test cases also cover essential issues like user authentication, data flow between the user interface and the database, and third-party integration, with an emphasis on banking services. This covers testing for proper data collection, API interfaces, error handling, security, and financial regulatory compliance. The goal is to ensure that the system maintains user data and interactions with external banking APIs in a secure and efficient manner, especially for sensitive tasks such as creating bank accounts.

4. Demonstrate (preferably a table) traceability between the test cases/plan and user story

User Story ID	User Story Description	Test Case ID	Test Case Description	Expected Outcome
US001	User Registration	TC001	Verify that users can successfully register by providing required details.	Successful registration redirects user to a confirmation page.
US002	User Login	TC002	Confirm that registered users can log in with correct credentials.	Successful login redirects user to the dashboard.
US003	Document Capture	TC003	Test the capability to capture and upload documents.	Application allows users to upload various document formats. Also guied which documents required to upload.
US004	ML Data Extraction	TC004	Ensure the ML model accurately extracts key information from uploaded images.	Extract the text from document where the relevent information is there as per the model trained in key value pair. Extracted data matches the content of the uploaded document.
US005	Fill Web Page Fields	TC005	Verify that the application correctly populates web page fields with extracted information.	All relevant fields are filled with accurate data in form as per third party requirement.
US006	Data Verification and Correction	TC006	Confirm that users can verify and correct filled information before submission.	User can edit and correct any inaccurate information.
US007	Form Submission	TC007	Test the submission process after users verify and correct the information.	Successful submission redirects to a confirmation page.
US008	Secure Database Storage	TC008	Ensure user data is stored in the database with encryption, tokenization, and masking.	Data in the database is encrypted, tokenized, and masked.
US009	View and Update Personal Information	TC009	Verify that users can view and update their stored personal information.	User can see and modify their information as needed.
US010	Digital Signature for Data Consent	TC010	Test the digital signature functionality for data consent.	Successfully signing the digital signature records consent in the database.
US011	Admin Account Management	TC011	Confirm that the admin can manage user accounts, third-party accounts, and user agreements.	Admin dashboard allows for account management and agreement tracking.
US012	Customer Support	TC012	Test admin support for any software malfunction or user issues.	Admin support resolves issues and provides assistance if user faces any.
US013	Data Privacy and Storage	TC013	Verify admin's ability to handle data privacy and storage policies.	Admin can configure and manage data privacy settings.
US014	Third-Party Integration	TC014	Confirm integration with third-party websites for filling required fields.	Successful integration with third party webiste like bank allows user to fill required fields with text extracted from documents.
US015	Accurate Data Extraction and Storage	TC015	Ensure accurate extraction of important data from uploaded images and secure storage in Azure Cloud.	Extracted data is precise, and stored data is secured in Azure Cloud Storage.

In the comprehensive test case table provided for the system verification phase of the web application, various critical functionalities were thoroughly examined to ensure the robustness and reliability of the software. The first set of test cases (TC001 to TC002) addressed user registration and login processes, confirming that users could successfully register and log in, leading to the expected outcomes of successful redirects and access to the dashboard. Subsequently, document capture and ML data extraction were scrutinized (TC003 to TC004), confirming the application's capability to upload various document formats and accurately extract essential information from images. Test cases TC005 to TC007 ensured that the application appropriately filled web page fields with extracted data, and allowed users to verify, correct, and submit forms, with expected outcomes leading to successful submissions and redirects. The following test cases (TC008 to TC010) delved into securing database storage, allowing users to view and update personal information,

and implementing digital signatures for data consent, verifying encryption, tokenization, masking, and consent recording functionalities. Admin-related functionalities were then verified through test cases TC011 to TC013, ensuring efficient account management, customer support, and data privacy handling. Finally, the integration with third-party websites (TC014) and accurate data extraction and storage in Azure Cloud (TC015) were thoroughly tested to confirm seamless integration and secure data handling.

Overall, this comprehensive set of test cases methodically covered all 15 user stories, meticulously validating each aspect of the web application, from user interactions and document processing to data storage, security, and external integrations. The verification process aimed to ensure the application's functionality aligns with the specified requirements, providing a robust and user-friendly experience while maintaining the integrity and security of user data.

PART 4 REFLECTION

1. **Share the actual timeline for the execution of the project.**

Weeks 1-2 of Sprint 1: Project Initiation and User Management

Week 1: Establishing the Project and Preliminary Setup

- Implementing user registration and login: Provide a safe and easy-to-use login and registration process.
- Admin is in charge of User Account Management: in place the admin's user account management functionalities.

Week 2: Initial Discussions and Third-Party Investigation

- Investigating possible agreements with third parties
- Look into and assess contracts with possible outside partners.
- Start Talking About Integrations and Data Access: Get started talking about managing data access and integrating third-party services.

Weeks 3–4 of Sprint 2: UI prototyping and machine learning integration

Week 3: Configuring the Machine Learning Model - Detailed Analysis and ML Model Setup

- Install the selected machine learning model after a comprehensive analysis of the models.

Week 4: Software Challenges and UI Prototyping

- User Interface Prototyping and Feedback Gathering: Develop user interface prototypes; collect and evaluate user input.
- Solving the First Software Issues: Recognize and handle early software issues; Provide a workable solution.

Weeks 5–6 of Sprint 3: Enhancing User Interaction Features and Database Refinement

Week 5: Improving Database Management

- Improving Database Management Techniques: Optimize database management techniques to maximize effectiveness.

Week 6: Extraction of Images and User Data Interaction

- Initial User Data Visibility and Update Features Implementation: Include tools that let people see and edit their data.
- Starting the Extraction of Image Data and Overcoming Obstacles: Start removing obstacles and getting info from pictures.
- Combining User Input with UI Design Improvements: Apply user input to improve the user interface design.

Weeks 7-8 of Sprint 4: Admin Support, Cloud Optimization, Signatures, and Verification

Week 7: Investigating Digital Signatures and Validation

- Giving Features for User Data Correction and Verification Priority: Give priority to creating features that allow users to edit and validate their data.
- Examining and Activating Digital Signature Capabilities: Examine and start developing the capabilities for digital signatures.

Week 8: Cloud Optimization and Admin Assistance

- Enhancing Administrative Assistance Procedures: Enhance administrative support procedures to provide customers with better help.
- Thorough Analysis and Modification of Cloud Storage Options: Carefully assess and refine cloud storage options to ensure optimal efficiency and security of data.

2. Compare the proposed schedule with the actual timeline

Phase	Proposed Schedule	Actual Schedule
Weeks 1-2: User management and the start of the project	User Sign-up and Access	Sign-up and Login of Users
	An admin maintains accounts while a user records and uploads documents - User scans and uploads papers; exploring other options for data gathering; and Admin looks into agreements	Admin oversees initial setup and third-party agreements
	The administrator looks into third-party contracts and possible restrictions on the API and begin discussing.	Websites from third parties offer access and APIs.- The preliminary conversations around substitute third-party integration.
Weeks 3–4: Integration of Machine Learning	ML model gathers data by rendering through photos	Initial ML model research and considerations for alternative models
	Application loads content onto the page	User interface prototypes emphasizing intuitive design
	Admin assistance in the event that program malfunctions	Working together to solve problems related to software errors and investigating new features in light of early user comments
Weeks 5 and 6: User Interaction and Data Management	Users encrypt data before saving it in the database, etc	Refinement of database handling with an emphasis on data storage optimization.
	Users have access to and can update their personal data	Features with improved interactivity and user data visibility.
	Accurately extracts significant information from photos.	Recursive enhancements to image data extraction, tackling obstacles
	Completes the necessary fields with the extracted data.	User feedback sessions that are included into the UI design result in modifications

Weeks 7-8: Finalization and Data Verification	Users check and update data as necessary.	Features for user data correction and verification with improved user assistance
	To consent to data use, users digitally sign documents.	Thorough investigation and advancement of digital signature capabilities
	The administrator answers inquiries from clients.	Enhanced administrative support techniques for effective client support
	The administrator makes sure the Azure cloud stores data securely.	Utilized cutting-edge security protocols for Azure cloud data storage

3. **Describe and explain the deviation from the planned or proposed.**

Sprint 1: Project Initiation and User Administration

Deviation: User registration testing's scope was lowered.

Scheduled Test cases: (TC001) Test cases were planned to cover the login and user registration features.

Actual Implementation: We prioritized essential scenarios and limited the scope to (TC001) test cases due to time restrictions. This choice was made to uphold sprint deadlines and make sure the most important components were fully tested.

Sprint 2: UI prototyping and machine learning integration

Deviation: More Iterations of the UI Feedback

Scheduled Test Cases: Ten test cases for the UI prototype and user feedback.

Actual Implementation: We added (TC0012) test cases to the UI testing phase and incorporated more iterations of user feedback. This resulted in a longer testing period but also a more refined and intuitive user experience.

Sprint 3: Enhancing User Interface and Database Structure

Deviation: Delaying the Extraction of Image Data Testing.

Scheduled Test Cases: (TC0015) test cases, which cover the basic extraction of picture data and database management.

Actual Implementation: In order to assure data integrity, we pushed back testing for picture data extraction until the following sprint and concentrated on database-related test cases. This modification made it possible for later stages to conduct more thorough testing.

Sprint 4: Cloud Optimization, Admin Support, Signatures, and Verification

Deviation: Cloud Setup and Data storage

Scheduled Test Cases:(TC0013) test cases that address cloud optimization, admin support procedures, and user data verification.

Actual Implementation: In order to handle unforeseen difficulties, we expanded the testing step for digital signatures to (TC0013)test cases. Prior to integration into the live environment, this decision was made with the intention of guaranteeing the security and dependability of the digital signature feature.

These modifications were developed to strike a compromise between comprehensive testing and the real-time restrictions of each sprint, taking into account the dynamics of the project. The team's ability to maximize testing efforts and provide a reliable product was made possible by constant observation and adjustment.

Sprint	Durantion
Sprint 1	10/1/23 - 10/15/23
Sprint 2	10/15/23 - 10/29/23
Sprint 3	10/29/23 - 11/13/23
Sprint 4	11/13/23 - 11/27/23

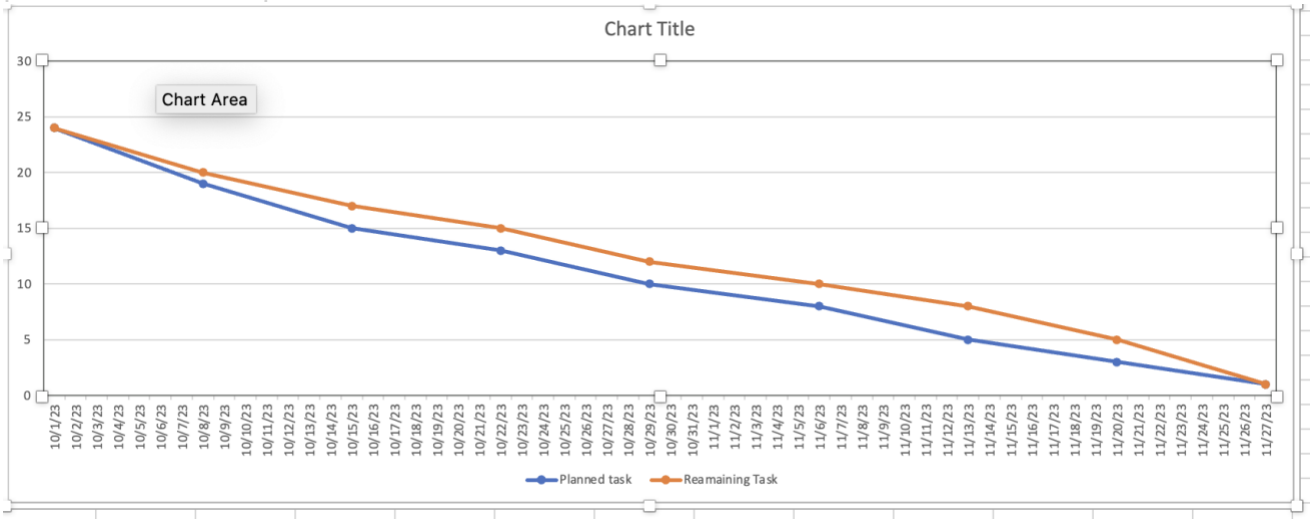

Figure 8: Burn chart of Sprint Deviation

References

S. Surana, K. Pathak, M. Gagnani, V. Shrivastava, M. T. R and S. Madhuri G, (2022). "Text Extraction and Detection from Images using Machine Learning Techniques: A Research Review," 2022 International Conference on Electronics and Renewable Systems (ICEARS), Tuticorin, India, 2022, pp. 1201-1207, doi: 10.1109/ICEARS53579.2022.9752274.

C. Kaundilya, D. Chawla and Y. Chopra, "Automated Text Extraction from Images using OCR System," 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2019, pp. 145-150.

Bern, A., Pasi, S.J., & Smolander, K. (2007). Contextual Factors Affecting the Software Development Process – An Initial View.

Lamberti, A (2021, Sept). How to extract text from any image with Deep Learning, from https://medium.com/artificialis/how-to-extract-text-from-any-image-with-deeplearning-e834d5a9863e

Wikipedia contributors. (2023, August 8). EF English Proficiency Index. In *Wikipedia, The Free Encyclopedia*. Retrieved 20:57, September 5, 2023, from https://en.wikipedia.org/w/index.php?title=EF_English_Proficiency_Index&oldid=1169375114

Software engineering: Software design - javatpoint. www.javatpoint.com. (n.d.-a). https://www.javatpoint.com/software-engineering-software-design

Cooper, A., Reimann, R., & Cronin, D. (2007). About face 3: The essentials of interaction design. Wiley

Intelligent diagramming. Lucidchart. (n.d.). https://www.lucidchart.com/pages/

pCloudy. (2022, March 4). *Verification and validation in testing*. Medium. https://pcloudy.medium.com/verification-and-validation-in-testing-

41f1aeab4651#:~:text=Verification%20is%20more%20of%20an,the%20reliability%20of%20the%20product

Desikan, S., & Ramesh, G. (2008). Software Testing: Principles and Practices. Publisher.

Humble, J., & Farley, D. (Year of publication). Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation. Publisher.

Bishop, C. M. (2012). Pattern Recognition and Machine Learning. Publisher.