SPMP - Data Extraction & Visualization of Form-Like Structured Documents

Darshan Satra, Nikhil Sharma, Param Shendekar, Vishal Salgond

1st October, 2021

Contents

1 Introduction

1.1 Project Overview

Various enterprises have massive amounts of scanned forms that need to be processed every month and added to the enterprise database.The data is also used for generating insights for the organization. All this needs a lot of time and effort in manual data entry into the computer system and then further analysis. This involves a lot of manual repetitive tasks which take up a lot of time and energy. In enterprises the data that is present in the form is mostly printed text apart from the signature, this gives us a window to utilize the advancements in the area of OCR, Natural Language Processing, etc to deliver a product that will automate the manual, cumbersome, error-prone process and generate insights with the data that we have received.

Using text extraction technique, machine learning and data visualization we intend to provide a solution which automates the manual data entry by extracting information from the uploaded forms and store it in the database of the organization. In addition, we will also be using sentiment analysis on relevant fields and using data analysis to produce meaningful insights for the organization and visualize it using data visualization techniques.

Expected Delivery Date: **April 2022**

1.2 Project Deliverables

- Software Requirement Specification - 01/10/2021
- Software Project Management Plan - 01/10/2021
- Wire frame Design - 06/10/2021
- Software Design Document - 08/10/2021
- Software Test Document - 15/10/2021
- Synopsis - 05/11/2021
- Source Code - April, 2022

2 Project Organization

2.1 Software Process Model

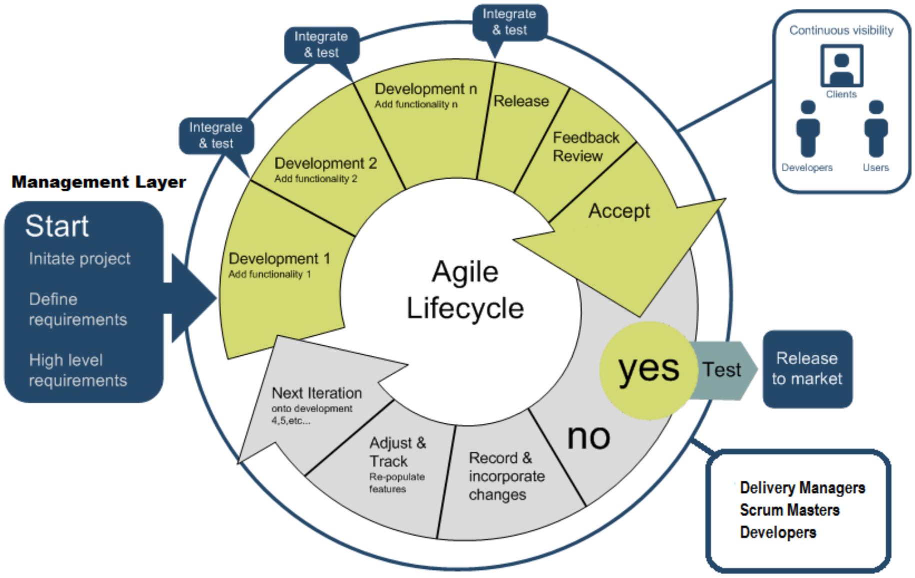


Figure 1: Agile Lifecycle

Agile software development is more than frameworks such as Scrum, Extreme Programming, or Feature-Driven Development (FDD).

Agile software development is more than practices such as pair programming, test-driven development, stand-ups, planning sessions, and sprints. Agile software development is an umbrella term for a set of frameworks and practices based on the values and principles expressed in the Manifesto for Agile Software Development and the 12 Principles behind it. When you approach software development in a particular manner, it's generally good to live by these values and principles and use them to help figure out the right things to do given your particular context.

We will have the following versions:

- **Beta:** This will be an MVP (minimum viable product) and will be completed by the end of seventh semester. With the beta, we will be able to check the validity of our product and get proof of concept.
- **Version 1:** This version will include a production-ready environment with most of the major features implemented using API's and will be given out

for review to investors to get feedback on the application.

- **Version 2:** This version will include the all the full-fledged features and the features suggested after reviewing the Beta version and version 1.

2.2 Roles and Responsibilities

Role	Responsibility	Team Member
Project Manager	Keeping track of all aspects of the project and timely execution and completion of project deliverables.	Param Shendekar
Analyst	Finding out the scope, practicality and required market research for the project and understanding the requirement in depth.	Darshan Satra, Nikhil Sharma
Developer	Involves creating the project by coding the required features either from scratch or building using the available APIs.	Darshan Satra, Nikhil Sharma, Param Shendekar, Vishal Salgond
Designer	Work hand in hand with the developer in designing the UI and UX of the application.	Vishal Salgond
Tester	To perform various tests and check the proper functionality of the application made by the developer.	Darshan Satra, Nikhil Sharma, Param Shendekar, Vishal Salgond

2.3 Tools and Techniques

Following are the Tools and Techniques to be used in the entire course of the project:-

1. **Programming Languages:** Python, JavaScript
2. **Backend:** Django
3. **Libraries:** Tesseract, Tensorflow
4. **Frontend:** React.js, HTML, CSS, JavaScript, Bootstrap
5. **Version Control:** Git, GitHub
6. **Databases:** MongoDB

3 Project Management Plan

3.1 Tasks

3.1.1 Task 1: Requirement Analysis and Creating Documentation

3.1.1.1 Description A complete analysis of the requirements would be done according to the deliverables mentioned and decided. The documentation mentioned would involve the SRS, SPMP, SDD, STD, Synopsis. These documentations would help us lay down a basic foundation for the project. By doing so, all the members of the project team would be on the same page, and there will exist a clarity about way of working and moving forward.

3.1.1.2 Deliverables and Milestones By the end of this process, the final revised versions of the SRS, SPMP, SDD, STD and Synopsis document would be delivered. The characteristics and functioning of the aforementioned platform would be clear to the design and development team. These deliverables would act as a major support in development and presentations.

3.1.1.3 Resources Needed Discussions with the client for determining the requirements and technical limitations of the application. A thorough literature survey would act as a valuable resource in finalizing the needs and requirements before hand.

3.1.1.4 Dependencies and Constraints The accuracy of these documentation is crucial to the project so it cannot be prepared because the entire workflow depends on the plan made in these documents. They act as a blueprint and guiding light whilst development and testing phases. The constraint should be that the language used should be clear, specific and understandable.

3.1.1.5 Risks and Contingencies Changes in client requirements and project budget can potentially delay as well as modify the existing flow of project due to but natural changes in the documentation.

3.1.2 Task 2: User Interface Design

3.1.2.1 Description The User Interface (UI) is the point of human-computer interaction and communication in a device. A well designed User Interface (UI) is imperative for a good user experience as it determines how users will interact with the application to access its functionalities. This includes the different screens of the application and the design of various elements in each of the screens.

3.1.2.2 Deliverables and Milestones The final version of User Interface design and delivery of a full-fledged user interface for users before the deadline.

3.1.2.3 Resources Needed Access to UI and Wireframe design tools such as Adobe XD and Whimsical.

3.1.2.4 Dependencies and Constraints The UI should be user-friendly and all the visualizations should be clearly visible and expandable. The user interface highly depends on the final version of all the features decided by the client.

3.1.2.5 Risks and Contingencies Poor or complex design of the User Interface may pose difficulties for clients in accessing the website.

3.1.3 Task 3: Text Extraction Module

3.1.3.1 Description This task will primarily focus on detecting the fields and extracting the key value pairs out of those fields from a form. This is a important feature, which will help us extract data and do some analysis on those extracted data.

3.1.3.2 Deliverables and Milestones By the end of this process, we will have a complete working algorithm to extract the data out of the submitted form. This algorithm will further be used in backend for processing purpose.

3.1.3.3 Resources Needed Good amount of research papers and online resources to build the algorithm and get the text extraction process working.

3.1.3.4 Dependencies and Constraints The input form should be of the type PNG or JPEG, with size not more than 4MB. The types of fields that will be detected will be limited because of our limitation with the dataset. This module will provide the further development in the task-4

3.1.3.5 Risks and Contingencies The algorithm might give very low accuracy result which will give the wrong insights about the form data.

3.1.4 Task 4: Back-end Implementation Module

3.1.4.1 Description The main focus of this module will be on providing API for the front-end for various tasks such as Authentication, project creation, form upload, etc. Here we will also integrate the text extraction model and use it to extract text and store it in the database.

3.1.4.2 Deliverables and Milestones The end deliverables of this module will be a fully functioning user management system, database schema creation, form upload functionality and text extraction model utilization. The whole process will be rolling as more models get developed they will get integrated to the backend.

3.1.4.3 Resources Needed Access to wireframes to design the API's response and the text extraction model. This module should function properly and complete mediation should be provided.

3.1.4.4 Dependencies and Constraints The text extraction module will be implemented only if the text extraction model is working properly. The responses of the API can not be finalized until the wireframes are completed. If the text extraction model is not functioning properly then the data inconsistency will arise. If the model is not flexible or requires very specific type of input then some amount of image processing will have to be done in the backend. If the wireframes are not accurate and they do not represent the screens that the actual app has then the backend api's response will need to be redone.

3.1.4.5 Risks and Contingencies The response time is slow and processing data takes a lot of time, then it might lead to bottlenecks and network chocking. It will hamper the scalability of the system and it will also lead to a bad user experience. If complete mediation is not provided then data inconsistency might arise. If user authentication module is not implemented correctly then security risks will arise.

3.1.5 Task 5: Sentiment Analysis Module

3.1.5.1 Description Sentiment analysis is the process of detecting positive or negative sentiment in text. This module will include the research and implementation of the sentiment analysis model and backend integration of the same.

3.1.5.2 Deliverables and Milestones After the implementation of this module our platform will be able to perform sentiment analysis on relevant fields. This will allow us to gain valuable insights and the output will be used by the Data Analysis module.

3.1.5.3 Resources Needed Relevant dataset to train the model. The required output format to give the results in right format. This module will depend on relevant python frameworks.

3.1.5.4 Dependencies and Constraints The output and the accuracy of the sentiment analysis module high depends on the type of input and the amount of data that is available.

3.1.5.5 Risks and Contingencies If the model does not have sufficient amount of data for training (which can happen if there aren't sufficient number of forms available in a particular project) then the output of the model can be wrong.

3.1.6 Task 6: Data Analysis and Visualization Module

3.1.6.1 Description Once the user forms has been process and the required texts has been extracted, our system will store all those data in our database. The data store will be used for different types of analysis. The platform is built such a way, that the user with no manual work can visualize all the submitted forms. The visualization will help user gain insights about the form-data.

3.1.6.2 Deliverables and Milestones We will have a production ready platform, in which the user can submit their forms to gain insights.

3.1.6.3 Resources Needed The main resource upon which this module resides upon is the data extracted in the Text Extraction Module and the one which is processed and stored in Back-End Implementation Module. Additionally, the requirement of a visualization tool in language specific libraries will also be required.

3.1.6.4 Dependencies and Constraints The Data Analysis & Visualization Module would be highly dependent on the cleaned, clear data received and stored in the MongoDB database. The database would then be queried for data particular to that project and it should be preprocesses in such a manner that it could be fed to the libraries which help in data visualization and insight generation.

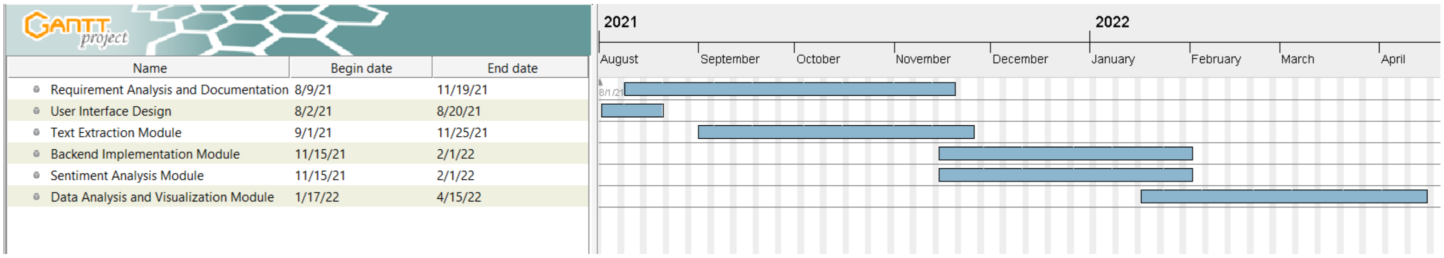
3.1.6.5 Risks and Contingencies The development might take more time than expected because of some technical difficulty. The responsiveness of the platform might hamper due to some charts and visualizations getting huge and varied quality of data. Also, the response time of the website whilst creating these charts might take a toll too.

3.2 Assignments

Sr. No	Task	Team Member
1	Requirement Analysis and Creating Documentation	Darshan, Nikhil, Param, Vishal
2	User Interface Design	Vishal
3	Text Extraction Module	Param, Nikhil
4	Back-End Implementation Module	Vishal, Param
5	Sentiment Analysis Module	Darshan, Nikhil, Vishal
6	Data Analysis and Visualization Module	Darshan, Nikhil, Param

3.3 Time Table

Sr. No	Task	Start Date	End Date
1	Requirement Analysis and Creating Documentation	09/08/2021	20/11/2021
2	User Interface Design	01/08/2021	20/08/2021
3	Text Extraction Module	01/09/2021	25/11/2021
4	Back-End Implementation Module	15/11/2021	01/02/2022
5	Sentiment Analysis Module	15/11/2021	01/02/2022
6	Data Analysis and Visualization Module	15/01/2022	15/04/2022



Name	Begin date	End date
Requirement Analysis and Documentation	8/9/21	11/19/21
User Interface Design	8/2/21	8/20/21
Text Extraction Module	9/1/21	11/25/21
Backend Implementation Module	11/15/21	2/1/22
Sentiment Analysis Module	11/15/21	2/1/22
Data Analysis and Visualization Module	1/17/22	4/15/22

Figure 2: Gantt Chart