

# Software Design Documentation

**Project Name : Text-Tools**

Date: 2 NOV 2002

Written By: KARTIK HATWAR

## Introduction

Text-Tools combine powerful tools related to text and languages and bring them together for you. Text-Tools can help every individual with some daily text related specific tasks.

## System Overview

**Text-Tools have five main features :**

1. **Image to Text Translator** : This powerful tool extract text from images , user need not have to download image from anywhere to convert , user can just put image url and get extracted text from it.
2. **Summary Tool** : With the help of this tool user can summarise long paragraphs into short and meaningful summary.
3. **Language Translator** : Language Translator helps user to translate the text to different languages and save them.
4. **Question Answer Tools** : This tool is bit tricky , it takes your question and figure out possible answers for it , some answers maybe irrelevant to your question but most of time user get his task completed with this tool.
5. **Text to Speech Converter** : For those who don't want to read long paragraphs or essays , this tool converts these texts into audio file which user can listen comfortably anytime.

With all these features User can do various task , also all the results of users queries can be save in the user account.

## Design Consideration

**Assumptions and Dependencies:**

Text-Tools took help from various AI and Machine Learning API's to do its task and it uses firebase to store user data.

**Goals and Guidelines**

Main Goal of Text-Tools is to help users in their daily text related task such as assignments , homework , translation , reading long paragraphs , etc.

**Guidelines :**

- User should have Password of minimum 6 letters otherwise Sign-Up page will not proceed anything.
- User should have Image Link to upload Images in Tools.

## System Architecture

---

An overview of how the functionality of the system were divided to subsystem

- In the Front-End, EJS(Embedded JavaScript) Templates are used to achieve high reusability which helps in dividing the page components in various sub components using EJS Partials.
- Navigation Bar, Footer, Services, About Page Partials [Templates] are created to maintain clean and tidy main page code, also as Navigation Bar and Footer are used in every page of Site, creating their templates help to achieve high reusability and save time.
- For Back-End, NodeJS and ExpressJS are used, to optimise code pages some common features such as file downloading function, getting links from firebase, saving file to store, saving text are created as functions and used many times in code.
- **Basically Every Function is divided into Three parts in the Back-End -**
  - 1) Whenever User goto page link [GET Request], function will provide the page to user.
  - 2) When User uses the page function [POST Request], function takes the user input parameters, uses them to implement specific API features, take the output from API and convert it in readable/audible format and show it to User.
  - 3) When User want to save output which user got from feature, function take output and save it to user specific path in Firebase.

## API USAGE

---

We used API in Backend with the help of NodeJS, as it have several advantages like

- When user login in Text-Tools, user's private data which was not intended to show will stay in the server and is inaccessible to browser thus user private data is safe.
- Some APIs need different or customised input from users basic input, so Back-End have additional flexibility to convert user's input into desired input and serve it to APIs.
- As all API functioning is processing from server, browser will have less load and can work faster as browser doesn't have to fetch any data from external APIs.
- Some Times API can give errors or not desired results as intended, so for error handling, backend can deal with server the proper data to user doesn't get any error.