

International Journal of System of Systems Engineering

ISSN online: 1748-068X - ISSN print: 1748-0671

<https://www.inderscience.com/ijssse>

GreenNotes: a smart utility tool to keep track of regular goals in the 21st century for smart cities and people

Rohit Rastogi, Karan Budhwani, Harsh Mittal

DOI: [10.1504/IJSSE.2023.10053192](https://doi.org/10.1504/IJSSE.2023.10053192)

Article History:

Received: 07 February 2022

Accepted: 09 May 2022

Published online: 16 February 2023

GreenNotes: a smart utility tool to keep track of regular goals in the 21st century for smart cities and people

**Rohit Rastogi*, Karan Budhwani
and Harsh Mittal**

Department of CSE,
ABES Engineering College,
Ghaziabad, UP, 201009, India
Email: rohitrastogi.shantikunj@gmail.com
Email: karan.18bcs1115@abes.ac.in
Email: harsh.18bcs1152@abes.ac.in

*Corresponding author

Abstract: GreenNotes is a browser extension that allows you to take notes rapidly. These notes may be used for a variety of things in your everyday life, like saving significant meeting discussion points, vital data for future reference, and so on. It allows you to make notes using a variety of techniques, including a WYSIWYG editor, a Markdown editor, and voice-to-text. This project seeks to create a note-taking Chrome extension with a reminder function that provides a convenient experience for users by giving rapid access and a simple UI. Current online solutions or programs either lack some functionality or are too pricey for those who only want basic services. GreenNotes, on the other hand, was created with freemium users in mind. GreenNotes is a Chrome extension built on Angular that allows you to take notes by typing or converting speech to text. Additionally, the app's reminder function was created to assist users in being on time for meetings or stand-up calls in the online context, which has become more prevalent since COVID-19.

Keywords: WYSIWYG; markdown; environment; end-user; extension; Google Chrome; plugin; manifest.

Reference to this paper should be made as follows: Rastogi, R., Budhwani, K. and Mittal, H. (2023) 'GreenNotes: a smart utility tool to keep track of regular goals in the 21st century for smart cities and people', *Int. J. System of Systems Engineering*, Vol. 13, No. 1, pp.1–29.

Biographical notes: Rohit Rastogi received his BE from C.S.S. Univ. Meerut, 2003; Master's degree in CSE from NITTTR-Chandigarh from Punjab University. He received his Doctoral degree from the reputed Dayalbagh Educational Institute, Agra, India. He is serving as Sr. Asst. Professor in the CSE Department of ABES Engineering College, Ghaziabad, India. He has won awards in several areas, including improved education, significant contributions, human value promotion, and long-term service. He keeps himself engaged in various competition events, activities, webinars, seminars, workshops, projects and various other educational learning forums.

Karan Budhwani is a student of BTech (CSE) in ABES Engineering College which is affiliated to Dr. APJ Abdul Kalam Technical University. He is keenly interested in Development. He is fond of traveling and meeting new people. He wants to explore and learn more in the domain of web development.

Harsh Mittal is a student of BTech (CSE) in ABES Engineering College which is affiliated to Dr. APJ Abdul Kalam Technical University. He is very enthusiastic about solving challenging problems. He is enamored with voyaging and meeting new individuals. He needs to investigate and learn more in the space of development.

1 Introduction

A browser extension is nothing more than a little piece of software. When software executes code, it is sometimes referred to as a ‘plugin’. This software unit applies numerous controls and filters to change how a user receives information coming from a web service such as an email, as well as how he reads webpages. It’s essentially a piece of code that enhances the browsing experience.

1.1 Extensions and its utility

Extensions are small apps that allow you to personalise your browser experience. They let users to customise Chrome’s functionality and behaviour in a variety of ways, including:

- instruments for increasing productivity
- enriching the content of a website
- information gathering
- games and amusement.

The Google Chrome online shop has a large number of Chrome extensions with unique features. Unlike apps or websites, these are not designed to accomplish a lot of work in a short period of time. Instead, they are goods that focus on certain features. Although some Chrome extension creation might be difficult, most of them are rather straightforward (Cloudiyapps et al., 2021).

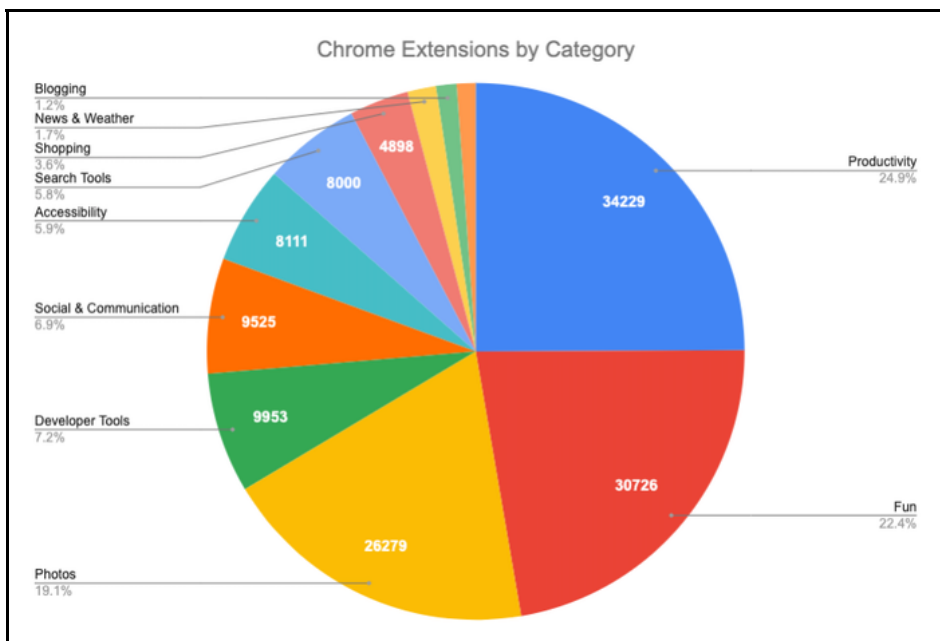
1.2 Various platforms/browsers for extensions

With the advent of the browser extensions API, the development of browser extensions became more consistent. However, the API implementations and extent of coverage change amongst the browsers that support the extensions API. The major platforms that support browser extensions are Chrome, Edge, Brave, Firefox, Opera, and Safari.

1.3 Global stats for extension

- In the year 2020, the total count of items on the Chrome Web Store was 2,15,075, out of which 1,55,583 were chrome extensions.
- Out of these extensions, approximately 50% have been downloaded less than sixteen times.
- Around 88% of all the extensions have an active user count of less than 1000.
- The most famous category of chrome extensions is **Productivity** with 46k + extensions.
- The second most famous is **Fun** with 33k + extensions.
- About 45% of the chrome extensions have not been updated by their authors for more than 2 years.
- Around 10% extensions have not been updated even once.
- Surprisingly, around 13% of the extensions have been downloaded only once by the users.
- About 50k extensions have a 5-star rating (TechACake et al., 2020).
- The following pie chart represents the percentage-wise weightage of different genres of chrome extensions available on the chrome web store (Figure 1).

Figure 1 Pie chart representing the available chrome extensions by category (see online version for colours)



1.4 *Software project management and product perspective of extension utility*

- Efforts have been taken to fully and precisely clarify the requirements.
- SRS defines product deliverables, product viewpoint, product activity, user characteristics, general restrictions, assumptions, functional requirements, external interface requirements, hardware and software requirements, and different design constraints.
- The user needs for this project have been divided into several parts.
- As a result, the system has been split. Figure 13 depicts the use-case diagram (Peters, 2020).
- There is only one type of user in this system.
- Because each user is only allowed to execute particular activities, the diagram depicts the system's various operations.

1.5 *Notes making and reminder feature in high tech. Smart City LifeStyle*

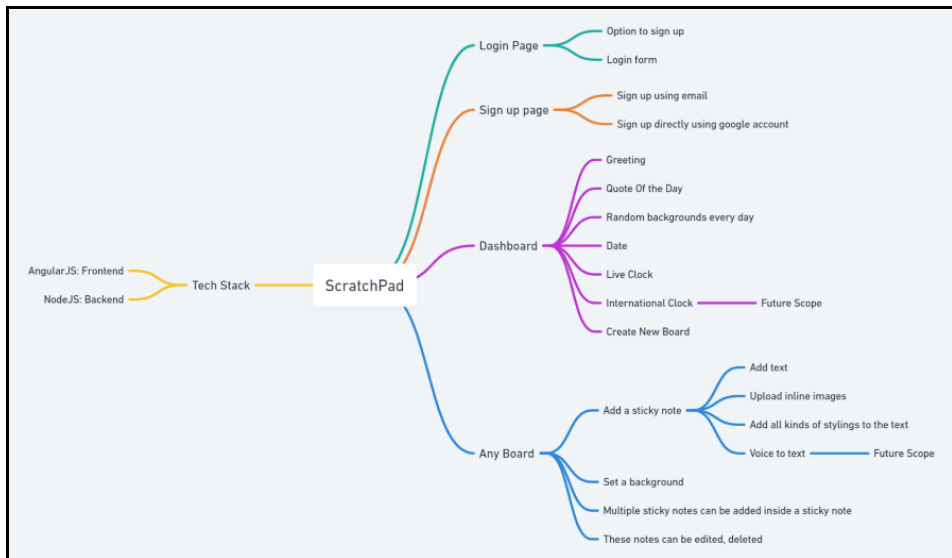
- GreenNotes helps its users to create, store and organise the notes as per their requirement or preferences.
- The voice-to-text feature of GreenNotes makes it more user interactive.
- It reduces the tedious job of typing long paragraphs.
- The reminder feature of the GreenNotes further increases its utility for an additional purpose.
- It makes sure that the user is always on time for their meetings and appointments.
- GreenNotes is overall designed by keeping in mind the modern day usages of each & every individual, specialising during the pandemic where most of the target sectors have shifted to online setting (Vinay, 2021).
- Additionally, GreenNotes reduces the use of pen-paper based notes-taking practices, thus contributing in green management (Gavade et al., 2021).

1.6 *Mind map*

A mind map is an outline used to outwardly coordinate data. A mind map is variously levelled and shows connections among bits of the entirety.

In the above mind map, the modularity of the GreenNotes is depicted and also depicts the functionality provided through each module (Figure 2).

A 'system' is the combination of elements that function together to produce the capability required to meet a need. The elements include all hardware, software, equipment, facilities, personnel, processes, and procedures needed for this purpose; that is, all things required to produce system-level results.

Figure 2 Mind map of GreenNotes (see online version for colours)

2 Literature review

Content editing represents the kind of PC application that takes advantage of a PC's capacity to store and show information, handle unique and complex information structures and respond promptly in order. Most of flow research on collaboration procedures has zeroed in on techniques for direct control, voice and signal contribution as the prime methods of associating, a Virtual Notepad was proposed which is an assortment of point of interaction devices that permits the client to take notes, comment on archives and information text utilising a pen, while still drenched in virtual conditions.

The current solutions available are either less user friendly or have a premium service only, that too at a high amount. Many of them only provide plain text editors.

GreenNotes is not just a typical text-in particular (plain text) editor. The subsequent records are regularly saved on cloud, making the program reasonable for altering framework documents to use in a DOS climate and, at times, source code for later arrangement or execution, for the most part through an order brief. It is moreover valuable for its immaterial utilisation of framework assets, making for fast burden time and handling time, particularly on under-controlled equipment (Murage, 2020).

GreenNotes acknowledges text from the clipboard. At the point when clipboard information with various configurations is glued into GreenNotes, the program just acknowledges text in the Markdown design. This is valuable for stripping implanted text style type and style codes from organised text, like when replicating text from a site page and gluing it into an email message or other WYSIWYG word processor. Organised text can be briefly glued into GreenNotes, and afterward promptly replicated again in stripped configuration to glue into the other program.

2.1 *Different apps with notes making feature*

The finest note-taking apps make it simple to jot down ideas, business insights, and even reminders while on the road. These more complex apps can take a variety of shapes, ranging from attempting to duplicate the word processing experience to simulating blank notepads in digital form, allowing for scribbling and sketching. Additionally, some note applications allow you to add photos, music, and even video to your notes, as well as share them directly with coworkers and teams.

1 Notepad

- It is a chrome extension and it is available on the Chrome Web Store. It runs offline and has already 2lakh + users using it.
- It provides a very simple UI with only one single editor to save notes.
- It works offline.
- It provides functionality to create multiple notes, delete and reorder these notes.
- It also provides a feature to download your note (<https://chrome.google.com/webstore/detail/notepad/ffbhefmlcoihbjcmibbfkocmnaiacinp?hl=en>).

2 Standard Notes

- It also provides a single source to store all your notes at one place and sync them across different devices.
- It works on a freemium pricing strategy, meaning it gives some basic features like creating plain text notes, but has monthly/yearly charges for premium features like creating notes using Modern editors.
- It works online and provides storage to store notes over the cloud.
- You can delete and add notes to bin using this tool.
- Its free version does not have web support, and only provides support for mobile and desktop applications (<https://standardnotes.com/>).

3 Gedit

- It is a plain text editor and comes pre-installed with the GNOME desktop environment. It was initially released 1999 and is written in C and Python.
- It has a very clean and simplistic GUI and provides a plain text editor.
- It is free and works offline.
- It also supports syntax highlighting for various languages (Python, HTML, XML, Javascript and many others)
- Multiple tabs are supported (<https://wiki.gnome.org/Apps/Gedit>).

4 Synotepad

- It is also one of the many chrome extensions that works as a notepad. It has a total of 585 users and has a 4.5 rating on chrome webstore.

- It has a very basic UI.
- Users can add multiple notes and delete them.
- It stores data in local storage.
- Only a plain text area is available for normal text type notes (<https://chrome.google.com/webstore/detail/synotepad/mmmmbiihnjgnaklgnpjnpffehfilhpndo>)

5 Rapid Table

- It is a web app that provides a plain text editor. It is rated as the top tool to create notes online in some papers (<https://allthatsaas.com/roundup/best-free-online-notepads/>, <https://productivityland.com/free-online-notepads/>)
- It has the very least interactive UI among all stated tools.
- It is free and stores data over the cloud.
- Users can only add one note, i.e., if a user wants to create a second note, he/she will have to delete the first one.
- Users can also open text files from the system (<https://www.rapidtables.com/tools/notepad.html>).

3 Methodology and setup of experiment

The procedures or strategies used to find, select, process, and analyse information about a topic are referred to as methodology. The following methodology section is designed to help the reader assess the project's overall validity and reliability.

3.1 Name of technologies used

- *Angular*

Angular is a TypeScript-based programming platform. Angular includes the following features as a platform:

- A framework for creating scalable web applications based on components.
- A set of well-integrated libraries that cover a wide range of functions, such as routing, form management, client-server communication, and more.
- A set of developer tools that aid you in the development, building, testing, and updating of your code (<https://angular.io/docs>)

- *Express*

Express is a Node.js web application framework that offers a comprehensive range of functionality for both web and mobile apps.

Express adds a thin layer of basic web application functionality without obscuring the Node.js capabilities you already know and appreciate.

- *Typescript*

TypeScript is a JavaScript-based strongly typed programming language that provides improved tooling at any size.

To support a stronger interaction with your editor, TypeScript adds additional syntax to JavaScript. In your editor, catch errors early. TypeScript code is converted to JavaScript, which can be executed anywhere JavaScript is supported: In a browser, on Node.js or Deno, and in your applications. TypeScript comprehends JavaScript and employs type inference to provide powerful tooling without the need for additional code.

- *Bcrypt*

It is a third party library used at server side to encrypt user password before saving it to the database.

bcrypt is a password-hashing function developed by Niels Provos and David Mazières and first presented at USENIX in 1999. Apart from integrating a salt to guard against rainbow table attacks, bcrypt is an adaptive function: the iteration count may be increased over time to make it slower, making it resistant to brute-force search attempts even as computing power increases.

- *Ngx-Quill*

ngx-quill is an angular module that contains all of the components you'll need for the Quill Rich Text Editor.

Quill is a modern WYSIWYG editor that is free and open source. It is entirely configurable to match any demand thanks to its modular architecture and expressive API.

3.2 *Types of databases*

A database is an ordered collection of data that is stored and retrieved electronically in computing. Large databases are housed on computer clusters or cloud storage, whereas small databases can be kept on a file system. Data modelling, efficient data representation and storage, query languages, security and privacy of sensitive data, and distributed computing challenges such as providing concurrent access and fault tolerance are all part of database architecture.

A database management system (DBMS) is software that captures and analyses data through interacting with end users, applications, and the database itself. The DBMS software also includes the essential tools for managing the database. A database system is the combination of the database, the DBMS, and the accompanying applications. The term 'database' is frequently misused to refer to any of the DBMS, the database system, or a database-related application.

There are two kinds of databases being used in GreenNotes:

- *PostgreSQL*

PostgreSQL is a sophisticated, open source object-relational database system that has a solid reputation for stability, feature robustness, and performance after more than 30 years of active development.

- *IndexedDB*

IndexedDB is a low-level API for storing large volumes of structured data, such as files and blobs, on the client side. This API makes use of indexes to allow for fast data searches. Web Storage is good for keeping little quantities of data, but it's not so good for storing big volumes of structured data. IndexedDB is a service that can help. This is the homepage for MDN's IndexedDB coverage; here you'll find links to the entire API reference and usage guidelines, as well as information on browser compatibility and an explanation of essential concepts.

3.3 Hardware requirement

The following represents the hardware system requirements in a precise manner (Figure 3).

Figure 3 Representation of system requirements

System requirements

Requirement	Minimum	Recommended
RAM	2 GB of free RAM	4 GB of total system RAM
CPU	Any modern CPU	Multi-core CPU. DataGrip supports multithreading for different operations and processes making it faster the more CPU cores it can use.
Disk space	500 MB and 1 GB for caches	SSD drive with at least 5 GB of free space
Monitor resolution	1024x768	1920x1080
Operating system	Officially released 64-bit versions of the following: <ul style="list-style-type: none"> • Microsoft Windows 8 or later • macOS 10.14 or later • Any Linux distribution that supports Gnome, KDE, or Unity DE. Pre-release versions are not supported.	Latest 64-bit version of Windows, macOS, or Linux (for example, Debian, Ubuntu, or RHEL)

Before you install and use our browser extension, make sure your computer meets the minimal system requirements.

- *Windows*

You'll need Windows 7, Windows 8, Windows 8.1, Windows 10 or later to use GreenNotes on Windows.

Intel Pentium 4 or later processor with SSE3 capability.

- *Mac*

You'll need the following to use GreenNotes on a Mac:

El Capitan 10.11 or later (Mac OS X)

- *Linux*

You'll need 64-bit Ubuntu 18.04+, Debian 10+, openSUSE 15.2+, or Fedora Linux 32 + to use GreenNotes on Linux.

Intel Pentium 4 or later processor with SSE3 capability.

3.4 Software requirement

For GreenNotes, the bare minimum software requirements are as follows:

- *Google Chrome Browser*

Google Chrome Browser is required by the developer as well as a regular user to run GreenNotes. Although other browsers also supports such extensions, but Google Chrome provides the best development as well as user experience

- *IDE*

Any IDE, like VSCode, JetBrains Webstorm, etc., can be used for the development purpose. An IDE, or Integrated Development Environment, allows programmers to combine the many parts of building a computer program into one convenient location. IDEs boost programmer productivity by merging typical software development tasks such as editing source code, creating executables, and debugging into a single application.

- *Node environment*

Node.js is a scalable network application builder that uses an asynchronous event-driven JavaScript engine.

Node.js is influenced by and similar in design to Ruby's EventMachine and Python's Twisted. The event model is taken a step further in Node.js. An event loop is presented as a runtime construct rather than a library. In other systems, the event-loop is always started with a blocking call. Typically, behaviour is described using callbacks at the start of a script, and a server is started via a blocking function such as EventMachine::run at the conclusion (). There is no start-the-event-loop call in Node.js. After executing the input script, Node.js just enters the event loop. When there are no more callbacks to perform, Node.js exits the event loop. The event loop is concealed from the user, similar to browser JavaScript.

- *Express framework*

Express is a Node.js web application framework that offers a comprehensive collection of functionality for building web and mobile applications. It makes it easier to create Node-based Web apps quickly. Some of the core features of the Express framework are listed below.

- Allows middlewares to reply to HTTP requests to be put up.
- Defines a routing table for doing various actions based on HTTP Method and URL.
- Allows you to render HTML pages dynamically by supplying variables to templates.
- *Angular CLI*

The Angular CLI is a command-line interface tool for launching, developing, scaffolding, and maintaining Angular applications from a command shell.

It is a collection of TypeScript libraries and resources that you may add into your apps to practise topics of optional and primary functionality. The architecture is based on a few key Angular ideas.

NgModules, which provide a compilation environment for components, are the fundamental building blocks of the Angular programming language. These modules put together a functional group of related code.

A root module that enables bootstrapping is always present in an application, as are more feature module implementations.

- *Git Bash*

Git Bash is a program that allows you to use the Git command line on your operating system. It's a command-line shell that allows you to use git from the system's command line. A shell is a terminal program that allows you to interact with your operating system by typing commands.

- *Cloud Storage*

An estimated minimum of 10GB cloud storage is required to store the notes and images.

3.5 *Network requirement*

- A decent network connection is required for downloading and installing the extension from the web store.
- Furthermore, internet connection might be required to sign-up/sign-in and synchronise data with the cloud storage.

3.6 *OS requirement*

GreenNotes is an OS independent tool as it requires a browser to run on, which is easily available for all OSs.

3.7 *Steps of execution*

It is assumed that Node is installed in the target system.

- 1 Open a terminal inside the project directory.
- 2 Create a build of the project with the respective build command.

- 3 A folder named **dist** shall be visible inside the project root directory (Please refer Figure 4).
- 4 Open Google Chrome Browser.
- 5 Visit `chrome://extensions`.
- 6 Turn on the Developer mode switch on the top right corner (Figure 5).
- 7 Then click on the Load unpacked button (Figure 6).
- 8 Select the project build file inside the dist folder.
- 9 The extension shall now appear with other extensions added to the browser.

Figure 4 Directory structure showing the Dist folder (see online version for colours)

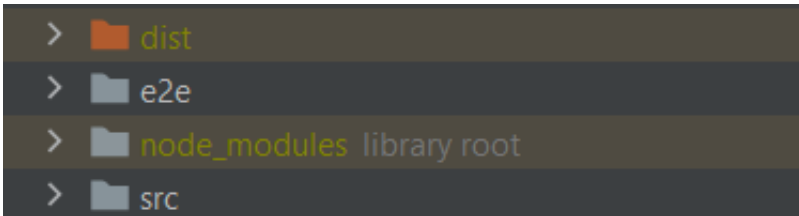


Figure 5 Banner showing the developer mode switch (see online version for colours)

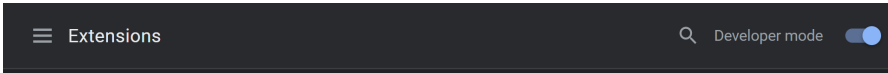
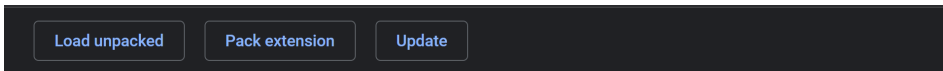


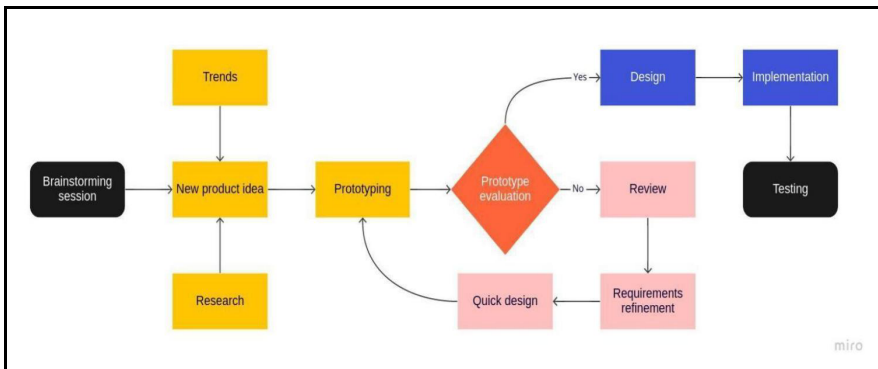
Figure 6 Banner showing the load unpacked button (see online version for colours)



3.8 Flow chart

This flowchart depicts the general flow of the complete development cycle, starting from brainstorming to implementation and testing (Figure 7)

Figure 7 Development cycle flowchart for GreenNotes (see online version for colours)

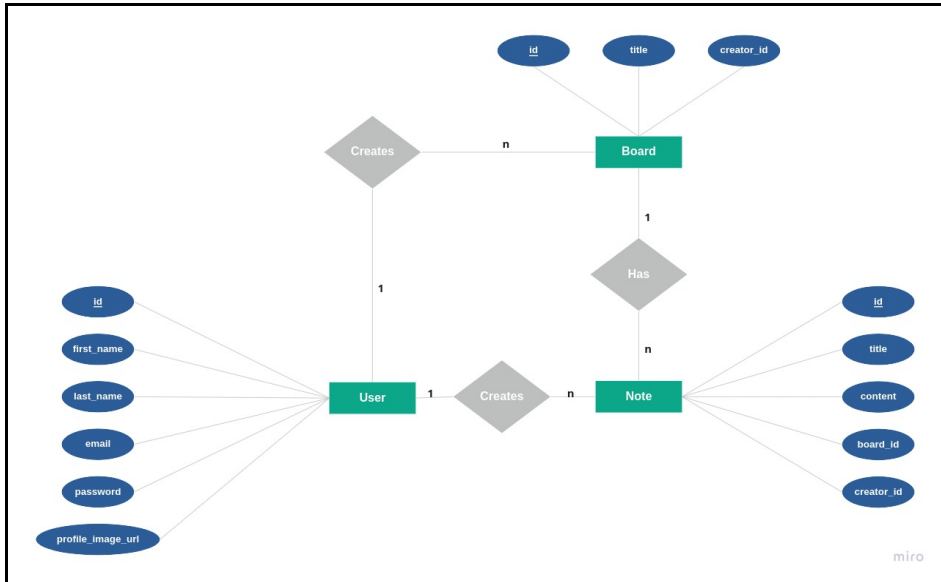


3.9 ER diagram

Entity Relationship Diagram (ER Diagram) is a diagram that shows the relationship between entity sets recorded in a database.

The following ER diagram represents the relation of the entities in GreenNotes (like Note, Board, User) and how they are stored in the database (Figure 8).

Figure 8 ER diagram of GreenNotes (see online version for colours)



3.10 Object oriented class diagram

A class diagram depicts the structure of a program by showing its classes, attributes, operations and relation with other classes.

The following class diagram showcases the major classes (User, Board, Note), their attributes and available functions. It shows the relation between User-Board and User-Note classes (Figure 9).

3.11 Data flow diagram (Level 0, Level 1, Level2)

DFD is basically the representation of a process or a system that depicts the flow of data.

The following data flow diagram depicts the interaction of the end-user with GreenNotes in three levels namely Zero, First, Second in order of increasing specificity. (Figures 10–12).

3.12 Use case diagram

A use case diagram depicts all the possible ways in which the use can interact with a system or a software.

Figure 9 Class diagram of GreenNotes (see online version for colours)

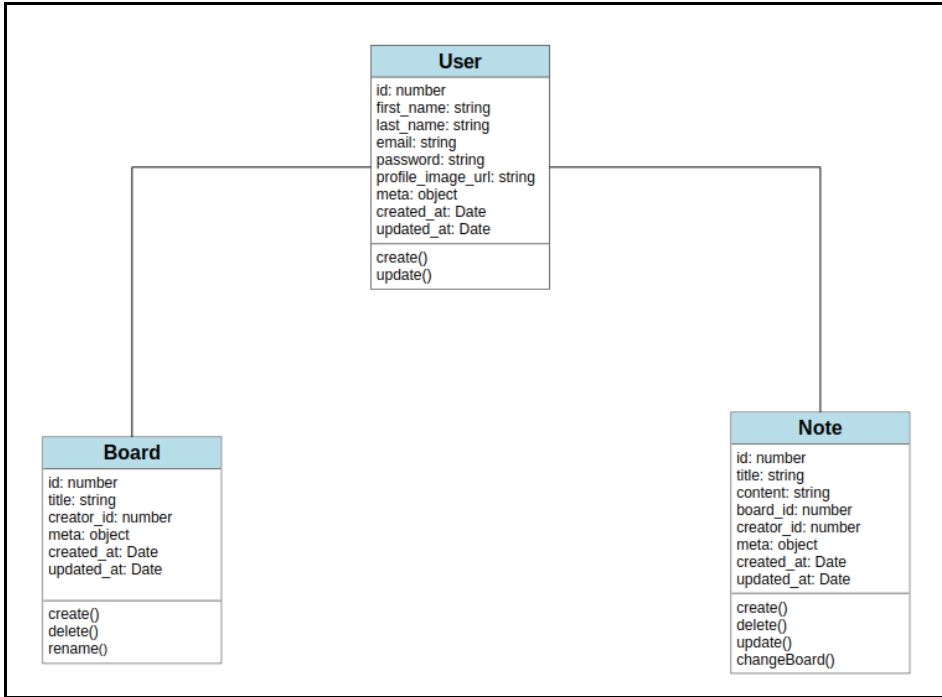


Figure 10 0 Level DFD of GreenNotes (see online version for colours)

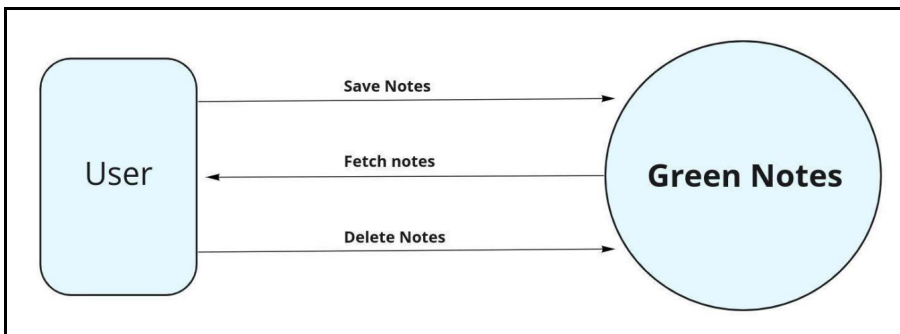


Figure 11 1st level DFD of GreenNotes (see online version for colours)

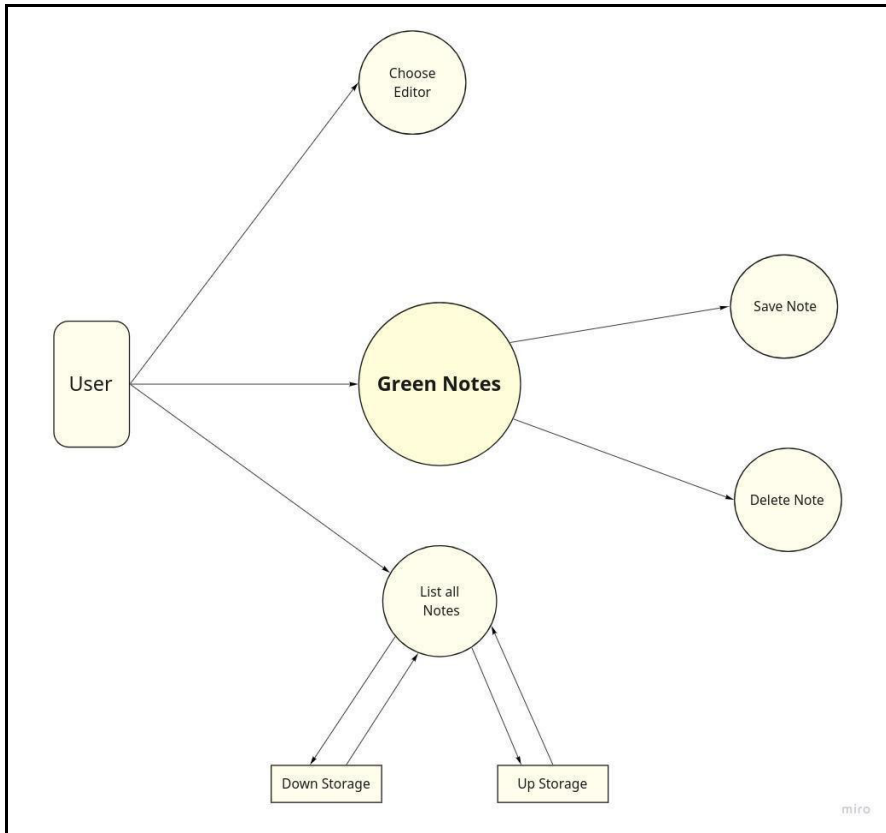
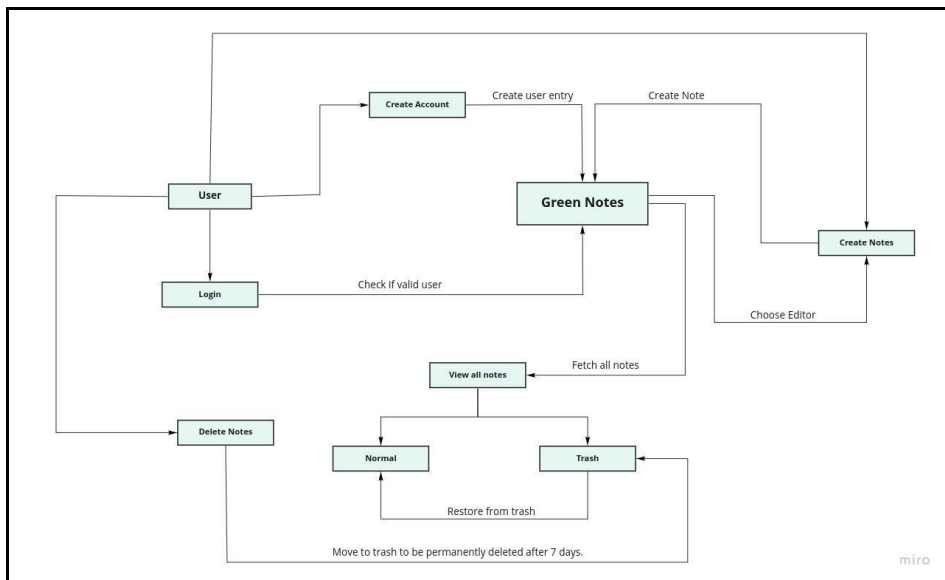
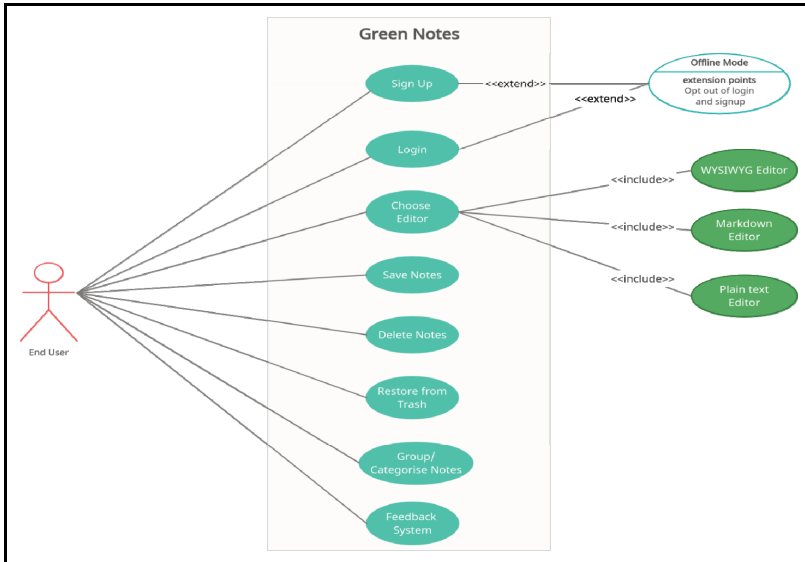


Figure 12 2nd level DFD of GreenNotes (see online version for colours)



In the above use case diagram, there is only one actor called End-User of our application. There are a total of 7 use cases in our app. Users can Sign Up, login, switch editors, save notes, and delete notes (Figure 13).

Figure 13 Use case diagram of GreenNotes (see online version for colours)

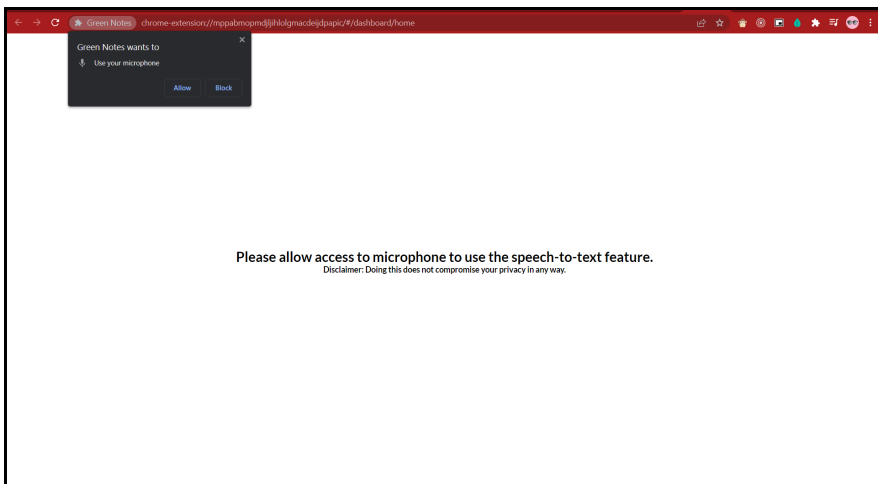


4 Results and discussions

- *GreenNotes asking for microphone access*

When GreenNotes is first installed, it asks the user for the microphone access, so that the user is able to use the smart speech-to-text feature of the GreenNotes (Figure 14).

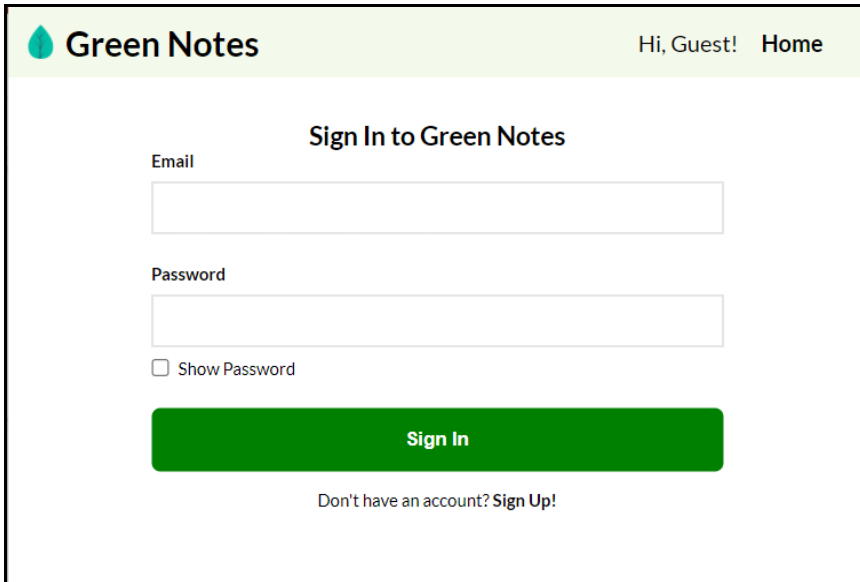
Figure 14 Permission page of GreenNotes (see online version for colours)



- *Login screen*

The login screen is a smart screen that enables the users to login to their GreenNotes account in order to store their data online or sync their data with cloud storage in order to avoid any data loss (Figure 15).

Figure 15 Login screen of GreenNotes (see online version for colours)

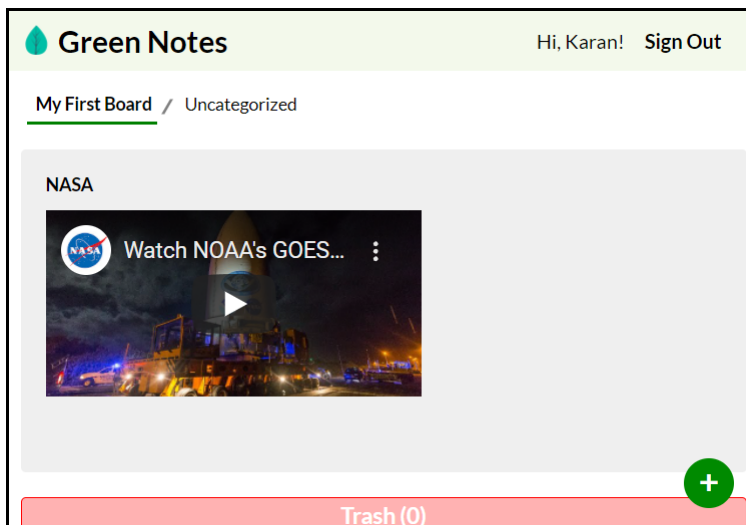


The screenshot shows the login interface for Green Notes. At the top left is the Green Notes logo (a green leaf) and the text "Green Notes". At the top right, it says "Hi, Guest!" and "Home". The main heading is "Sign In to Green Notes". Below this are two input fields: "Email" and "Password". Under the password field is a checkbox labeled "Show Password". A large green button with the text "Sign In" is positioned below the input fields. At the bottom center, there is a link that says "Don't have an account? Sign Up!".

- *Dashboard*

The dashboard is the landing screen of GreenNotes. This is where the user is able to see, create and manage their notes (Figure 16).

Figure 16 Main dashboard of GreenNotes (see online version for colours)

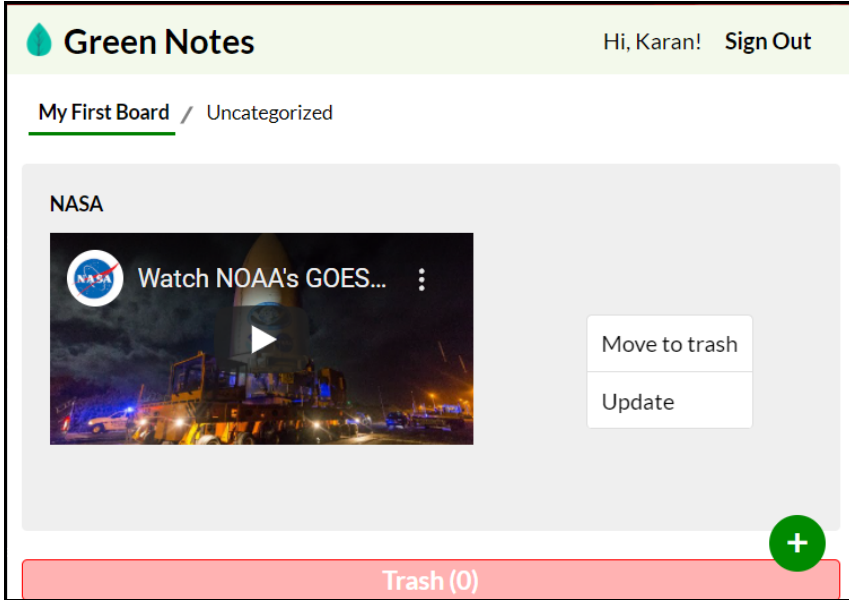


The screenshot shows the main dashboard of Green Notes. At the top left is the Green Notes logo and the text "Green Notes". At the top right, it says "Hi, Karan!" and "Sign Out". Below the header, there is a breadcrumb trail: "My First Board / Uncategorized". The main content area features a video player with the NASA logo and the text "Watch NOAA's GOES...". The video player shows a night scene with a large structure and lights. At the bottom right of the video player is a green plus sign icon. At the bottom of the dashboard, there is a red bar with the text "Trash (0)".

- *Note context menu*

Every note has its unique context-menu, that is triggered upon mouse right-click. This context menu has options useful for managing the note (Figure 17).

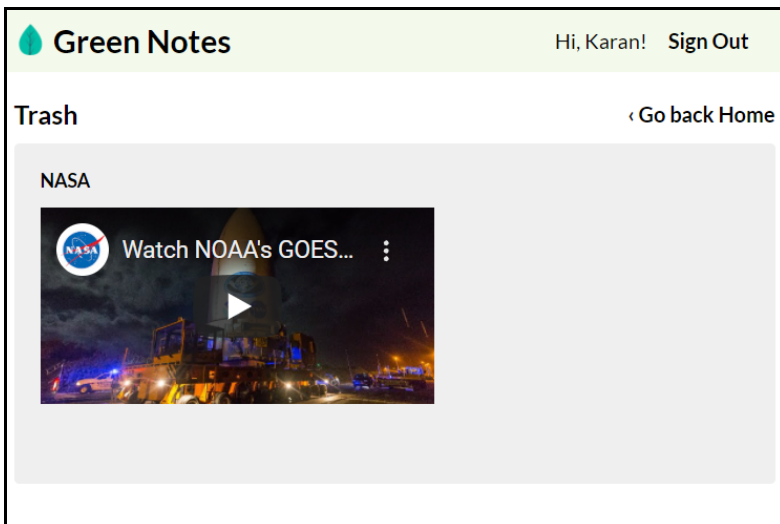
Figure 17 Context menu on note (see online version for colours)



- *Trash screen*

The trash screen contains the list of notes which have been soft-deleted, i.e, they have been removed from the dashboard, but still exist in the database (Figure 18).

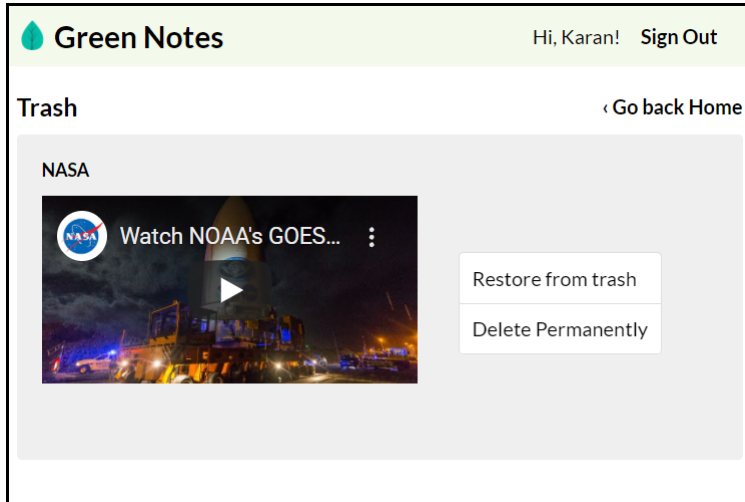
Figure 18 Green notes trash screen (see online version for colours)



- *Trash context menu*

The context menu of a trash note contains options to manage it. The user can restore the note or delete it permanently (Figure 19).

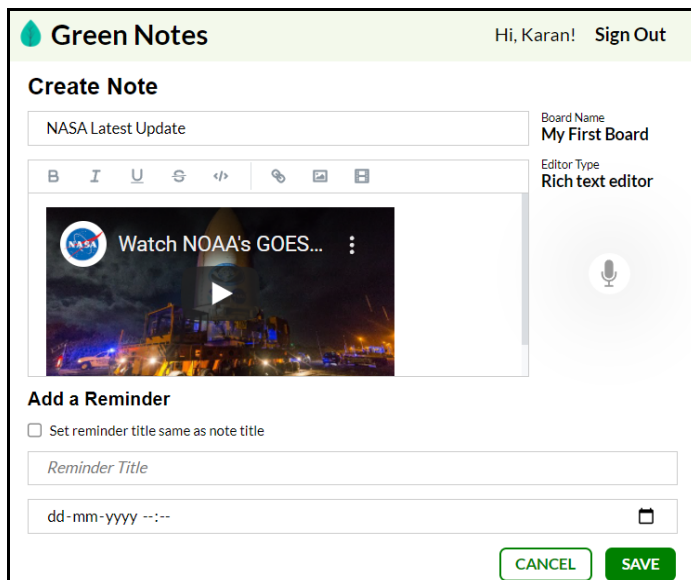
Figure 19 Context menu in trash (see online version for colours)



- *Create note screen*

Create Note Screen is the soul of GreenNotes. This is where the user can create a note, a reminder or both. The user can also use the microphone to use the speech-to-text feature (Figure 20).

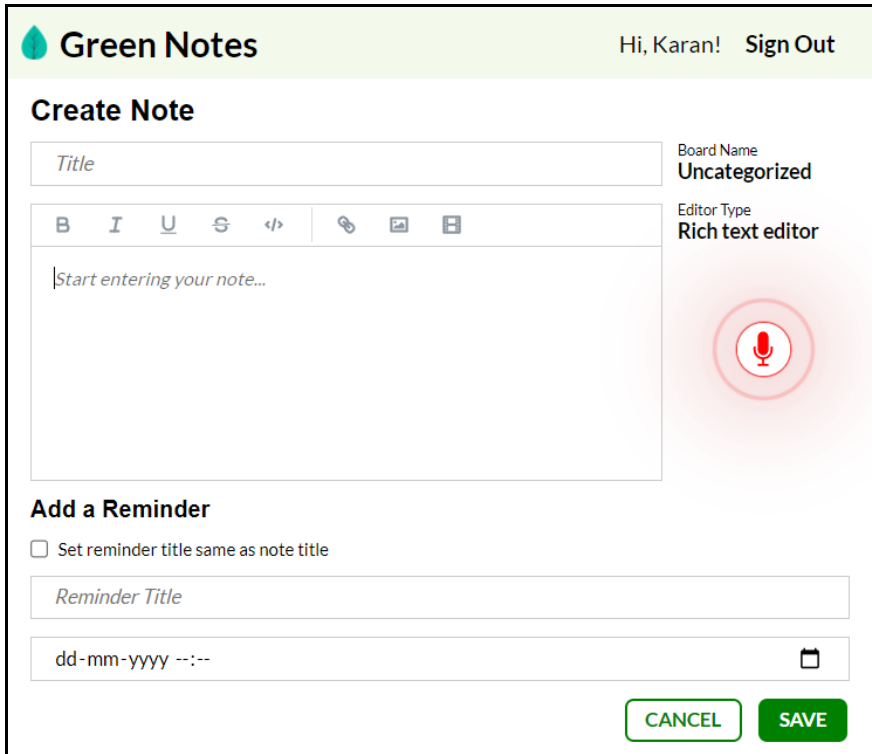
Figure 20 Create note screen of GreenNotes (see online version for colours)



- *Microphone in action*

Clicking once upon the microphone icon activates the speech recognition service and then whatever the user speaks can be seen updating into the editor in real-time. After using it, clicking it again will turn the microphone off (Figure 21).

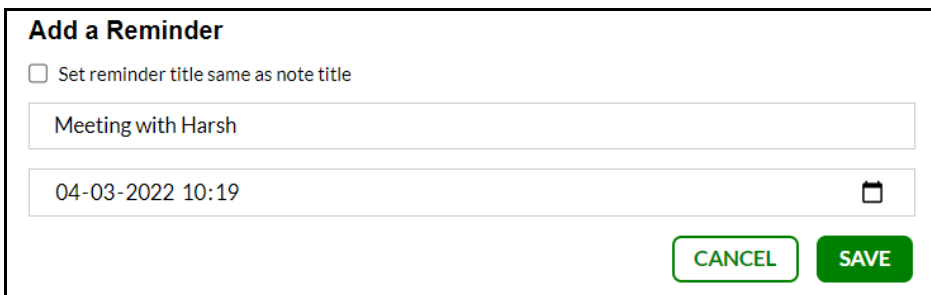
Figure 21 Create note using speech (see online version for colours)



- *Create reminder*

The user can create a reminder by selecting the date and time, and adding a title (Figure 22).

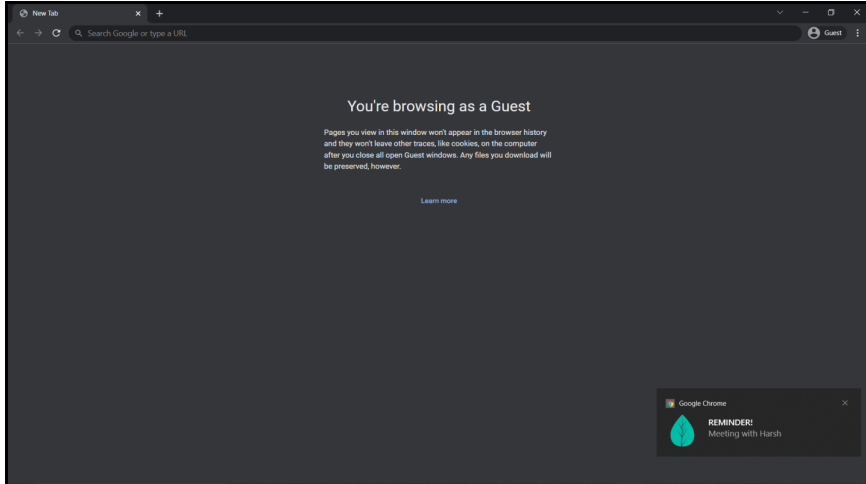
Figure 22 Create note with reminder (see online version for colours)



- *Reminder notification*

Once the selected time for reminder has elapsed, GreenNotes sends a notification to user (Figure 23).

Figure 23 Reminder notification with note title (see online version for colours)



- *App module*

The AppModule is the first module of the app that loads when the app is opened (Figure 24).

Figure 24 App module code snippet (see online version for colours)

```
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { AppComponent } from './app.component';
import { SharedModule } from './modules/shared/shared.module';
import { AppRoutingModule } from './app.routing.module';
import { CommonModule, HashLocationStrategy, LocationStrategy } from '@angular/common';
import { CoreModule } from './modules/core/core.module';
import { HttpClientModule } from '@angular/common/http';

const COMPONENTS = [
  AppComponent
];

const MODULES = [
  HttpClientModule,
  CommonModule,
  BrowserModule,
  AppRoutingModule,
  CoreModule,
  SharedModule
];

@NgModule({
  declarations: [...COMPONENTS],
  imports: [...MODULES],
  providers: [
    {provide: LocationStrategy, useClass: HashLocationStrategy}
  ],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

- *Package file*

The package.json file shows all the scripts required to interact with the project during the development phase, as well as the dependencies the project is using. (Figure 25)

Figure 25 Package.json code snippet (see online version for colours)

```
{
  "name": "green-notes",
  "version": "0.1.0",
  "scripts": {
    "ng": "ng",
    "start": "ng serve",
    "start:plugin": "ng build --watch",
    "build": "ng build",
    "test": "ng test",
    "lint": "ng lint",
    "e2e": "ng e2e"
  },
  "private": true,
  "dependencies": {
    "@angular/animations": "~11.0.1",
    "@angular/common": "~11.0.1",
    "@angular/compiler": "~11.0.1",
    "@angular/core": "~11.0.1",
    "@angular/forms": "~11.0.1",
    "@angular/platform-browser": "~11.0.1",
    "@angular/platform-browser-dynamic": "~11.0.1",
    "@angular/router": "~11.0.1",
    "@types/quill": "1.3.10",
    "ngx-quill": "13.0.1",
    "quill": "1.3.7",
    "rxjs": "~6.6.0",
    "tslib": "^2.0.0",
    "zone.js": "~0.10.2"
  }
}
```

- *Board model*

A model defines the constraints on an object. It depicts all the required and optional keys of the object of an entity. (Figure 26)

Figure 26 Board model code snippet (see online version for colours)

```
import {AutoIncrement, Column, DataType, Model, PrimaryKey, Table, Unique} from "sequelize-typescript";

@Table({ options: {
  timestamps: true,
  paranoid: false,
  tableName: "boards"
}})
export class Board extends Model {
  @Unique
  @PrimaryKey
  @AutoIncrement
  @Column(DataType.INTEGER)
  id: number;

  @Column(DataType.STRING)
  title: string;

  @Column(DataType.JSON)
  meta: JSON;


  @Column(DataType.DATE)
  createdAt: Date;

  @Column(DataType.DATE)
  updatedAt: Date;
}
```

5 Novelties

The above table shows the comparison of various similar software with the GreenNotes on the basis of features they provide (Table 1).

Table 1 Table depicting feature comparison between various note-taking apps (see online version for colours)

 Features Comparison					
Features Available	Notepad	Standard Notes	Gedit	Rapid Tables	Green Notes
Plain Text Editor	✓	✓	✓	✓	✓
WYSIWYG Editor	✓	\$	✗	✗	✓
Markdown Editor	✓	\$	✗	✗	✓
Store data over Cloud	✓	✓	✗	✓	✓
Sign In / Sign Up	✗	✓	✗	✗	✓
Speech To Note	✗	✗	✗	✗	✓
Multiple Notes	✓	✓	✓	✗	✓
Multiple Platforms Availability	✗	✓	✗	✗	✓

The green ticks mean that the feature is available, while on the contrary red crosses means that the feature is missing. The dollar(\$) sign means that the feature is available, but only for the premium users.

6 Recommendations

- GreenNotes is not less than a smart efficacy tool that has been precisely designed to make it less exhaustive for the users to create and manage the notes.

- It's clean and precise UI makes it very much user-friendly for all age groups.
- The future scope of GreenNotes intends to create Mobile App as well as Web App to access the data and also sync it between all these platforms.
- To increase the utility, the launch page can also provide minor details like World Clock (that can be set-up for any timezone), weather, etc that can affect its users in planning their day/schedule.

7 Future research directions and limitations

7.1 Limitations

- One of the limitations of GreenNotes is with IndexedDB. If someone with enough knowledge deletes any data from the IndexedDB, then it becomes a hefty job to recover it from the cloud, in case of logged in users, and is completely irretrievable in case of non-logged in users.
- In case of logged in users, if there are a large number of notes, and any one data entry is deleted from the IndexedDB, then the ids of all the notes will have to be compared with those available on the cloud. This will compromise the efficiency of the app.
- If instead a data entry from the boards table is deleted, then all the associated notes will be left dangling with no relation with any board.

7.2 Future directions

- The next research should be conducted in order to make the app the only truthy entity that can delete any data.
- Any loss in data that takes place outside the app should be made easily recoverable without any efficiency loss.
- GreenNotes can also be incorporated with other systems, to form a system of systems (SoS), to solve multiple problems at once. For example, GreenNotes can be integrated with process managers like PM2 for creating a note of the logs on hourly or daily basis for authorised users, thus reducing the hefty job of ssh-ing into the server every time.

8 Conclusions

- GreenNotes is a smart utility tool to keep track of regular goals in the 21st century for smart cities and people.
- With the advancement of online setting globally post pandemic, there has been a requirement of a tool that could help its users in planning and organising their day-to-day goals efficiently and flawlessly.

- GreenNotes, with its interactive design and efficient tools, promises to meet the daily usefulness criteria of the individuals.
- Also, the authors aim to extend the functionalities of GreenNotes under their future scope, which eventually will make GreenNotes one of a kind.

Motivation

- Every person who browses anything over the internet, at some point needs a place to store the data that they might feel will be useful in future.
- The currently available solutions in the market do not provide many features to organise notes/references to freemium users.
- Most of the users currently have to install separate apps to take notes, which might not be a good experience for most users, as maintaining lots of apps while working can make this whole process a little clumsy.
- Sometimes a user needs different editors (like WYSIWYG, Markdown, Plain) while taking notes to emphasise their points, which are hardly available in one single app.
- There are times when a user wants to keep some brief meeting notes, thus few applications have a speech to text option that encourages users to take notes by voice.

Scope of the study

- This project seeks to provide a highly scalable, user-friendly, and interactive solution for taking short notes without the need for many note-taking tools.
- To create a Chrome Extension, a Web App, and a Mobile App for taking notes, and to provide a single solution for all platforms and user preferences.
- Users should be able to register an account and save their data in the cloud, as well as sync their data across the mobile app and the web app using the same account.
- If a user does not want to make an account, he or she can create notes in their local storage, however those notes will be purged from their storage on a daily basis.
- This tool will allow three ways (editors) to write notes in the early stages of development:
 - Plain Text Editor
 - WYSIWYG
 - Markdown Editor

- This utility may be used entirely with a keyboard (for those who prefer less interaction with the mouse).
- Incorporate a mechanism that generates a fresh quotation every day (user can also turn this off from settings).

Topic organisations

This study gives an overview of how GreenNotes is a more promising software when compared to other software solutions of its genre that are available globally. The author team has explained about the various use-cases of the software that can affect the target users in their daily lives, by helping them planning and sorting their priorities and goals. For the endorsement of the study, the author team did a literature survey and reviewed five most downloaded softwares on the internet. The literature survey provides deep knowledge about the pros and cons of those software solutions.

The author team has further described the methodology in which they have represented the methods used for the study. The study uses technologies like Angular, Express, along with several software libraries and packages, for the development of GreenNotes. Further, the paper discusses the output of the project which is represented in the form of snapshots.

Suggestions for specific applications to solve the challenges and constraints discovered in the assessment have been offered in the recommendation section, which is one of the most essential aspects of the research study. The novelty section refers to aspects of the research that are novel. Finally, the conclusion section summarises the study's main findings and provides a final assessment.

Ethical Committee and Funding

The experiments do not include any human related experiments and so no ethical constraints have been violated. Though the subjects performing the study were humans and air quality directly affects them but the study does not violate any health related measures. The Project is not funded by any agency.

References

- Clouidyapps *et al.* (2021) *dev.to*, Importance of Chrome Extension Development, https://dev.to/cloudify_apps/why-do-your-businesses-need-an-api-driven-strategy-4k0a
- TechACake *et al.* (2020) *techacake.com*, Google Chrome extensions statistics, <https://techacake.com/chrome-extensions-statistics/>
- Peters, J. (2020) *dev.to*, Adding an Angular 9 Project as a Chrome Extension, <https://dev.to/jwp/adding-an-angular-project-as-a-chrome-extension-in-10-minutes-2hmp>
- Vinay (2021) *medium.com*, 6 steps to create a Chrome Extension using Angular, <https://medium.com/weekly-webtips/is-it-hard-to-create-a-chrome-extension-using-angular-d9fd6a5740f3>

- Murage, A. (2020) *section.io*, How to build and test a Node.js with Express, <https://www.section.io/engineering-education/building-a-basic-api-with-nodejs/>
- Gavade, S., Kulkarni, S., Mohite, S. and Bhole, K. (2021) ‘Development of integrated versatile paper, plastic and aluminum waste sorting and disposal system’, in Kumar, S. and Rajurkar, K.P. (Eds.): *Advances in Manufacturing Systems. Lecture Notes in Mechanical Engineering*, Springer, Singapore, https://doi.org/10.1007/978.981.33.4466.2_7

Websites

- Angular Documentation, <https://angular.io/docs>
- Chrome Extension Documentation, <https://developer.chrome.com/docs/extensions/mv3/getstarted/>
- Gedit, <https://wiki.gnome.org/Apps/Gedit>
- Node Documentation, <https://nodejs.org/en/>
- Notepad, <https://chrome.google.com/webstore/detail/notepad/ffbfefmlcoihbjcmibbbfkocmnaiacinp?hl=en>
- Rapidtables, <https://www.rapidtables.com/tools/notepad.html>
- Standard Notes, <https://standardnotes.com/>
- Synotepad, <https://chrome.google.com/webstore/detail/synotepad/mmmmbiihnjgnaklgnjnppfehfilhpnodo>

Annexure

A Additional Readings

- 1 What are chrome extensions?, <https://bit.ly/3nsSX4x>
- 2 Importance of Chrome extension, https://dev.to/cloudify_apps/why-do-your-businesses-need-an-api-driven-strategy-4k0a
- 3 Building Chrome Extensions, https://dev.to/cloudify_apps/why-do-your-businesses-need-an-api-driven-strategy-4k0a
- 4 Chrome Extension Boilerplate (Manifest V3), <https://dev.to/llagerlof/chrome-extension-boilerplate-manifest-v3-1mj>
- 5 What’s Really Going On In A Chrome Extension, <https://dev.to/singhshemona/what-s-really-going-on-in-a-chrome-extension-4na1>
- 6 Local Database and Chrome Extensions (IndexedDB), <https://dev.to/anobjectisa/local-database-and-chrome-extensions-indexeddb-36n>
- 7 How to run migrations in remotely database (Heroku Postgres), <https://dev.to/mariobatalha/how-to-run-migrations-in-remotely-database-heroku-postgres-4dki>
- 8 How to setup a Postgres Database with Heroku, <https://dev.to/tgrede/how-to-set-up-a-postgres-database-with-heroku-31i5>

B Key Terms and Definitions

- **Markdown:**

Markdown is a lightweight markup language for creating formatted text using a plain-text editor.

- **Environment:**

The term environment refers to the state of a computer, determined by a combination of software, basic hardware, and which programs are running. In a DOS system, the environment is an area in memory used by the operating system and programs to store various information.

- **End-user:**

An end user is a person or other entity that consumes or makes use of the goods or services produced by businesses. In this way, an end user may differ from a customer—since the entity or person that buys a product or service may not be the one who actually uses it.

C Supporting coding and result snapshots

Figure A Directory Structure of the App (see online version for colours)

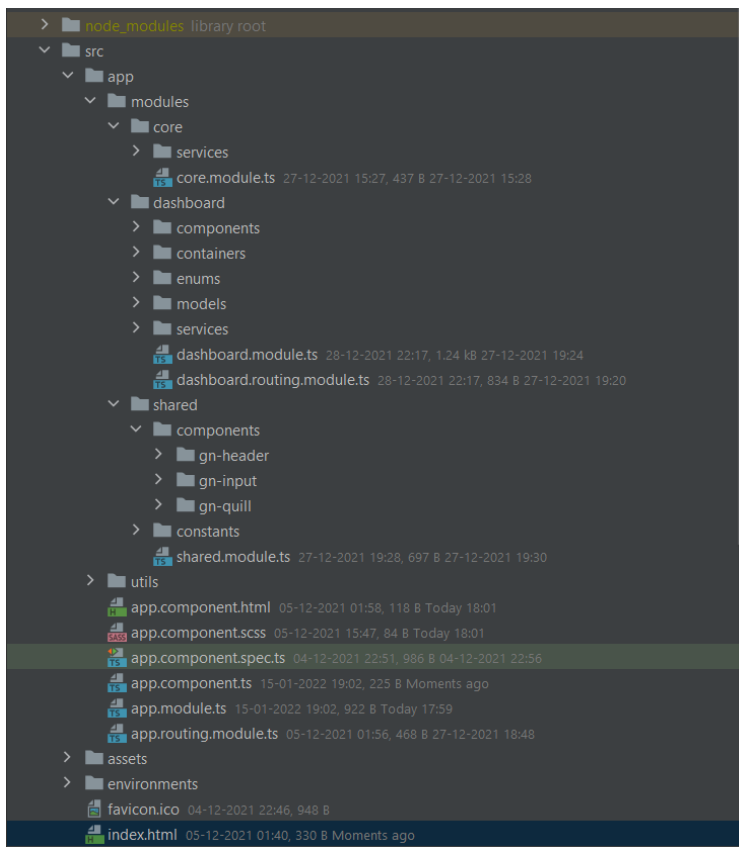


Figure B Dev Environment file of the App (see online version for colours)

```
export const BASE_URL = 'https://green-notes-dev.herokuapp.com';
```

Figure C tsconfig.json file (see online version for colours)

```
/* To learn more about this file see: https://angular.io/config/tsconfig. */
{
  "compileOnSave": false,
  "compilerOptions": {
    "baseUrl": "./",
    "outDir": "./dist/out-tsc",
    "sourceMap": true,
    "declaration": false,
    "downlevelIteration": true,
    "experimentalDecorators": true,
    "moduleResolution": "node",
    "importHelpers": true,
    "target": "es2015",
    "module": "es2020",
    "lib": [
      "es2018",
      "dom"
    ]
  }
}
```