

# Online Newspaper Delivery System

Maulina Raina  
201801186 - B.Tech. ICT  
DAIICT, Gandhinagar

Astha Shah  
201801169 - B.Tech. ICT  
DAIICT, Gandhinagar

**Mentor (on-campus):** Prof. JayPrakash Lalchandani  
DAIICT, Gandhinagar

**Abstract**—The project focuses on providing a better alternative for the existing newspaper delivery system that we are using currently. We have made use of our knowledge of web development and databases to implement and manage this system. The web application that we have made is named NewsDaily which will provide a better, efficient and user friendly interface to all its users and stakeholders so that the process of newspaper delivery can be made as smooth and error free as possible.

**Index Terms**—Database, User Story, Schema, Algorithm, API, Development

## I. INTRODUCTION

Reading newspapers is a routine in our life. Even in this digital era, 650 Million People read it every morning with their cup of tea. To get the newspapers delivered at your home, you have to contact a local newspaper delivery person using various methods. In this digital age where with one click, you can order your food, groceries, and many more items, doing all this work is not logical,

So we have developed a web application named NewsDaily, which can do all this work for you. All you have to do is login and signup and you can see a list of newspapers available in your city, your bill details, print them, and also view all the past orders. We have developed features for delivery personnel and vendor as well. As a delivery person, you will first add your profile. After that, you would be able to see the list of customers and details of the vendor. You can also maintain billing on the website. As a vendor, you will first add your profile and the list of newspapers that you sell. After that, you can see the list of the delivery person with their subscribed newspapers and quantity.

## II. PRE-PROJECT FIELD WORK

After doing quite some research on this topic on the internet and lots of discussion we concluded that some deeper insight was needed to understand the internal working of this system. So to understand if the ideas that we have were feasible and practical, we approached a local delivery boy and vendor. Talking to people who work in this sector showed us that we only knew the tip of the iceberg and a lot of things were still unknown. The delivery boy told us about his daily routine, the collection of newspapers, the transaction between him and the

vendor, the timings along with many other things. We asked all the questions and discussed some ideas that we had thought about the project and they helped us make a better and refined model, which made us realize the importance of digitizing the system and how it will be helpful to them.

## III. USER STORIES

The first step in our project was identifying users of the system and preparing user stories for each of them. There were four users in our system. Customer, Newspaper Delivery Person, Vendor, and Admin. Their roles were as follows: [1]

Login, sign up, profile creation, send issue/suggestions regarding system and logout - functionalities were for the customer, newspaper delivery person, and vendor.

- Customer requirements
  - So the main use of this system for customers is to view the newspaper available in their city and subscribe to the newspaper to get them daily at their doorstep.
  - Customers can view details of the bill, and past orders
  - Customers will get the notification on important updates
  - Customers can also apply for scrap service that will happen every month.
- Newspaper Delivery Person
  - The newspaper delivery person will enter his details in the profile section to provide his services. Then admin will view these details and take action on it.
  - On verification, he will get details of the vendor from whom he can buy newspapers for delivery purposes.
  - He will be able to see the list of customers that have been assigned to him by our system and collect the bill from them every month.
- Vendor
  - The vendor will enter his details in the profile section and give the list of newspapers that he will sell to provide his services. Then admin will view these details and take action on it.

- On verification, the vendor will see the list of newspaper delivery persons allocated to him by our system along with the quantity and cost associated with it.

- Admin

- The main purpose of the admin is to view details of the vendor and newspaper delivery person and take action on them.
- Admin can also view suggestions/issues and take action when needed.

#### IV. SYSTEM REQUIREMENTS

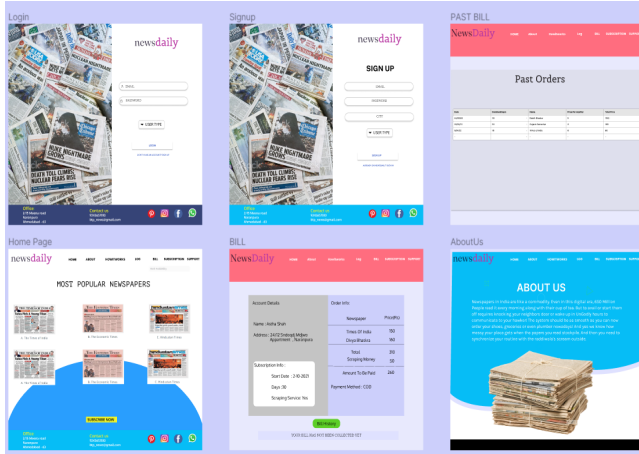
##### A. Tech Stack

- Front End : HTML , CSS , React [3][7]
- Back End : Node.js [2], Express.js
- Database : MySQL [6]

##### B. UI /UX

We used Figma to design all the templates for our web application. With the help of Figma, we were able to get an overview of our system before even coding it.

Here is the view of our Figma [5] [8]



#### V. SOLUTION DESIGN

##### A. Setup

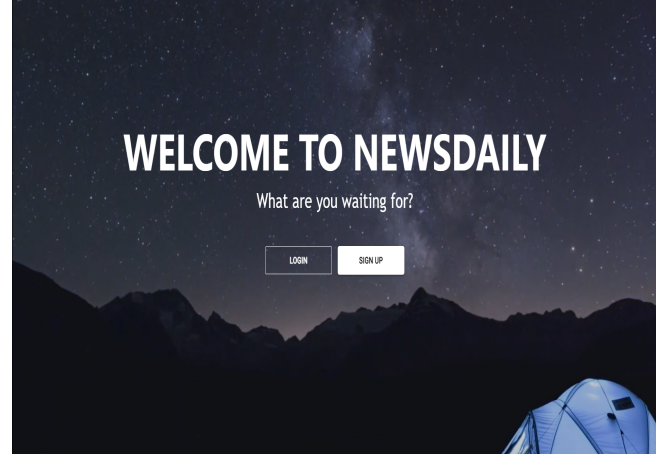
The initial setup of the system is done using GitHub. The installations of several packages and libraries were required such as react,node, dotenv, express and nodemon.

##### B. Run

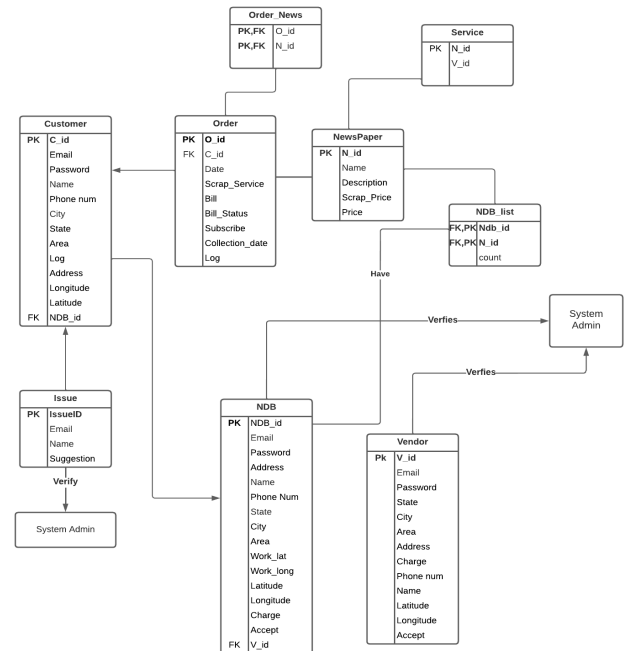
To run locally: Users have to clone the GitHub repository provided at the end of the report. Open the terminal and write "npm i" to install all the dependencies. Then write "npm start" to run the application. Follow these procedures for both front-end and back-end parts.

Apart from that, we have deployed our website on Heroku. So users can view our website by clicking on this URL: <https://newsdaily-btp.herokuapp.com/>

After running the application user will be able to see the Landing page as follows:



##### C. Database Schema



We have used MySQL for database management. Since it uses SQL, we have to define a proper schema. We have used ER diagrams to understand the relationships between tables. A list of tables with their details is given in Appendix.

#### VI. SYSTEM WORKFLOW

On opening the application, the first page that will be shown is the Landing Page. The landing page contains a suggestion box, information about each role, and buttons for login and signup. If you are a new user, then you have to sign up first by providing your email, password, state, city, and user type - customer, delivery person, vendor. After signing in, you will be redirected to different pages based on your role. From there you can view all the features available for that particular role.

- **Customer:** After the login and sign-up process in the application, the customer will be able to see all the

newspapers that are available in their area(if there are any). They can go read some basic information about the available newspapers to make the best choice among them. If the customer wishes to subscribe to the available newspapers, they can do that by pressing the subscribe button on the header. After clicking on that they will be navigated to the page where basic information like name, phone no, and address will be asked. After clicking on submit if a pop-up is shown which shows that service is not available then there are no available delivery people available in the given location so the process will stop until a delivery person is assigned that location. If the customer is navigated to the next page successfully, then they can proceed by choosing from the available newspapers and whether they would want to return them via the scraping process. After submitting that an invoice will be generated that can be viewed and downloaded by the customer. They can also view any past orders that they have done. About us page is provided to understand more about the website. A notification bell icon is also present on the header, there we will receive daily updates about the delivery of the newspaper and also a reminder about the subscription deadlines.

- **Newspaper Delivery Person:** After the login and sign-up process in the application, the delivery persons have to provide details in their profile to start delivering newspapers. Then admin will view these details and take action on it. If the admin accepts their requests, they will be able to see details of the vendor assigned to them. They can see the details of the customers and a list of newspapers subscribed by them. These lists will grow as more customers will join the application. Also, They have been given a button to send a notification. So when they deliver the newspaper to the customers, they can send the notification, so that their customers will get notified that the newspaper has been delivered. They can also view the list of the bills to collect from their customers and all the bills that they have been collected in the past month.
- **Vendor:** After the login and sign-up process in the application, the vendors have to provide details in their profiles. They have to give the list of the newspapers that they want to sell. Then admin will view these details and take action on it. If the admin accepts their requests, they will be able to sell the newspapers. They can see the details of the delivery persons and a list of newspapers subscribed by them. These lists will grow as more delivery persons will join the application.
- **Admin:** After putting the password on the login page for admins, the admin would be logged in. He/She would be able to see all the suggestions along with the emails put in by the users. Two more options of NDB(Newspaper Delivery Person) List and Vendor List will be present on the header. Clicking on either of them, a page would be shown of all the newspaper delivery persons and vendors respectively. The admin would be responsible to verify them by clicking the accept button shown with each

vendor and NDB. Only after their status is accepted they will be able to work with the system and use all the features provided to them.

In addition, we have added features like sorting through different labels, pagination, a search bar in all the pages to make it more easy to use and user friendly

## VII. ALGORITHM

When a customer subscribes, he will be assigned to the delivery person automatically. Similarly, when the newspaper delivery person creates a profile, he will get a vendor to buy the newspapers from. For that, we have developed an algorithm based for calculating minimum distance between set of points.

We are using latitude and longitude to calculate the distance. The formula is as follows :

```
function distance(lati_1 , long_1 , lati_2 , long_2)
{
    long_1 = long_1 * Math.PI / 180;
    long_2 = long_2 * Math.PI / 180;
    lati_1 = lati_1 * Math.PI / 180;
    lati_2 = lati_2 * Math.PI / 180;
    let dlong = long_2 - long_1;
    let dlati = lati_2 - lati_1;
    let vari = Math.pow(Math.sin(dlati / 2), 2)
        + Math.cos(lati_1) * Math.cos(lati_2)
        * Math.pow(Math.sin(dlong / 2), 2);

    let dist = 2 * Math.asin(Math.sqrt(vari));

    // Radius of earth in kms.

    let rad = 6371;

    // Final Result
    return(dist * rad);
}
```

### • Allocation of delivery person to customer :

In the Newspaper delivery person table, we have added a field called work\_lat and work\_long. The value of these fields is the latitude and longitude of the first customer that the delivery person has been assigned to. Now, the delivery person will get the customers within a two-km radius of that coordinates. Also, the first customer assigned to this delivery person will be less than 5 km distance from the home address of this delivery person. We are taking the minimum distance of the delivery persons (who have qualified the above criteria, either within a two-km radius from the workplace or within five km distance from the home address ) so that customers will get fast delivery. If we don't have any delivery person who satisfies the criteria given above, we will display the message that the service is not available to the customer.

### • Assigning vendor to delivery person

Here, the algorithm is simple. We are calculating the distance between the newspaper delivery person and

vendors. The delivery person will get the vendor who has a minimum distance. Also, the allocated vendor should not be far from 25km.

#### VIII. BENEFITS OF THE SYSTEM

- This system makes it easy for all the stakeholders to manage and keep a record of the stuff that goes on in the whole process.
- It gives a big advantage to the newspaper delivery person as rather than keeping a lot of information in memory or in writing they can now see all the necessary data anytime and anywhere at their convenience. system increases the efficiency and reduces the chances of human errors by the users like bill calculation, addresses, notifications about delivery, and many more.

#### IX. PROBLEM FACED

- **The API Problem** In our system, we have used an algorithm that automates the process of assigning customers to a newspaper delivery person and the newspaper delivery person to the vendor according to their nearest location.

Now, we needed an API that will do the work of getting the coordinates- latitude, and longitude of the location that the user has put in as their address, to be given to the algorithm for it to work.

The major problem that we faced here was finding the right API that satisfied all the requirements of the system and algorithm. Initially, our goal was to make this system for all the locations in India, but we could not do that with the APIs that we found in our research. We had only two requirements for the API- It should have rich and accurate data, especially for all the locations in India and we should be able to use it without paying any charge or putting in our credit card. But all the APIs which were accurate for India were not free, and which were free, they had not that accurate data for India.

Here accuracy is important as we are trying to assign the users based on the least distance, therefore we need the API to have at least that much accuracy to identify and give the coordinates of the correct street and building of the given address.

Example of an API that was good but paid-Google Maps. Examples of APIs which were free but not that accurate-Position Stack, MapBox, Yahoo Maps.

So finally we decided that right now for this system we would use the places and addresses of the USA by using the API of MapBox. The accuracy for the USA is quite high and our algorithm will be efficiently able to calculate the minimum distance we need here.

We would also like to continue our search for an API that can be used to retrieve the accurate coordinates for Indian areas as our algorithm is perfectly ready so just the input would change by this.

- **The Learning Hassle** This was just the initial delay that we had while starting the code portion of the project. As we did not have a lot of experience working with these languages so we initially dabbled in different languages trying to decide which one should we use here. So the learning and implementation phase simultaneously was a bit difficult in the beginning but then we gained momentum and it slowly became more fun to do it.

- **The Domain Problem** Initially, when we finalized this project, we thought that the information that we have gained from the internet would be enough for the system's proper implementation. But as we progressed and had more meetings and more discussions with our mentor we came to know that this domain is quite new and unknown to us and all the data and ideas cannot be searched on the internet.

It was quite difficult to meet the stakeholders because our timings did not match so we tried hard to set up a meeting with them. After a lot of tries, we finally set up a meeting and asked them all the relevant questions, doubts, and ideas that we had. So having an in-depth knowledge about the domain proved to be very valuable to us.

#### X. FUTURE SCOPE

- We are using the address of USA based region, we can do this for India as well. This will require an API that can provide accurate results for India.
- The notification feature can be made in real-time. So the user doesn't have to reload the page to get the notifications.
- Apart from that, the algorithm is developed using a naive approach. We can make it more efficient by considering other variables.

#### XI. ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my mentor for giving me the opportunity to learn technologies like react, node, express, and Figma and to work through the complete code of the application. It was a great experience to get hands-on with the whole cycle - from deep discussions about the system to the code of the complete application.

#### REFERENCES

- [1] <http://takeru2013.web.fc2.com/doc/srs.pdf>
- [2] <https://www.youtube.com/watch?v=Oe421EPjeBE>
- [3] <https://scrimba.com/dashboard#overview>
- [4] <https://mui.com/getting-started/installation/>
- [5] <https://www.youtube.com/watch?v=4W4LvJnNegA>
- [6] <https://www.w3schools.com/mysql/default.asp>
- [7] <https://reactjs.org/docs/getting-started.html>
- [8] <https://www.newspaperkart.com/>

## LINKS

- Figma:[https://www.figma.com/file/fVQMk55jZPtqJKOkHiFydi/BTP\\_Phase1?node-id=0%3A1](https://www.figma.com/file/fVQMk55jZPtqJKOkHiFydi/BTP_Phase1?node-id=0%3A1)
- GitHub:<https://github.com/astha9120/NewspaperDeliverySystem> [https://github.com/astha9120/NewsPaperSystem\\_Backend](https://github.com/astha9120/NewsPaperSystem_Backend)
- ER:[https://lucid.app/lucidchart/9c2c4beb-5e9d-4ff9-9706-a0e993be25eb/edit?beaconFlowId=F4275395D6E1BC97&invitationId=inv\\_75127cc3-a626-49f3-a1c9-80e4d9a8c3ef&page=0\\_0#](https://lucid.app/lucidchart/9c2c4beb-5e9d-4ff9-9706-a0e993be25eb/edit?beaconFlowId=F4275395D6E1BC97&invitationId=inv_75127cc3-a626-49f3-a1c9-80e4d9a8c3ef&page=0_0#)

## APPENDIX A DATABASE TABLES

TABLE I  
CUSTOMER

Field Name	Type	Nullable	Reference
c_id	int	No (PK)	NULL
email	varchar(255)	No	NULL
password	varchar(25)	No	NULL
state	varchar(255)	No	NULL
city	varchar(255)	No	NULL
area	varchar(255)	Yes	NULL
ndb_id	int	Yes	NDB
address	varchar(255)	Yes	NULL
name	varchar(255)	Yes	NULL
phoneno	varchar(255)	Yes	NULL
latitude	float(15,12)	Yes	NULL
longitude	float(15,12)	Yes	NULL
log	tinyint(1)	Yes	NULL

TABLE II  
NDB\_LIST

Field Name	Type	Nullable	Reference
ndb_id	int	No(PK)	ndb
n_id	int	No(PK)	Vendor

TABLE III  
ISSUE

Field Name	Type	Nullable	Reference
issue_id	int	No (PK)	NULL
suggestion	varchar(1000)	No	NULL
name	varchar(255)	Yes	NULL
email	varchar(30)	Yes	NULL

TABLE IV  
NEWSPAPER

Field Name	Type	Nullable	Reference
n_id	int	No(PK)	NULL
name	varchar(255)	No	NULL
description	varchar(1000)	No	NULL
scrap_price	float	No	NULL
price	float	No	NULL

TABLE V  
ORDERS

Field Name	Type	Nullable	Reference
o_id	int	No(PK)	NULL
c_id	int	Yes	Customer
date	date	Yes	NULL
scrap_service	tinyint(1)	Yes	NULL
bill	float	Yes	NULL
bill_status	tinyint(1)	Yes	NULL
subscribe	tinyint(1)	Yes	NULL
collection_date	date	Yes	NULL
log	tinyint(1)	Yes	NULL

TABLE VI  
ORDER\_NEWS

Field Name	Type	Nullable	Reference
o_id	int	No(PK)	Orders
n_id	int	No(PK)	Newspaper

TABLE VII  
VENDOR

Field Name	Type	Nullable	Reference
v_id	int	No(PK)	NULL
email	varchar(255)	No	NULL
password	varchar(255)	No	NULL
state	varchar(255)	No	NULL
city	varchar(255)	No	NULL
area	varchar(255)	Yes	NULL
charge	float	Yes	NULL
address	varchar(1000)	Yes	NULL
phoneno	varchar(30)	Yes	NULL
name	varchar(50)	Yes	NULL
latitude	float(15,12)	Yes	NULL
longitude	float(15,12)	Yes	NULL
accept	tinyint(1)	Yes	NULL

TABLE VIII  
SERVICE

Field Name	Type	Nullable	Reference
n_id	int	No(PK)	Newspaper
v_id	int	No(PK)	Vendor

TABLE IX  
NDB

Field Name	Type	Nullable	Reference
ndb_id	int	No(PK)	NULL
email	varchar(255)	No	NULL
password	varchar(255)	No	NULL
state	varchar(255)	No	NULL
city	varchar(255)	No	NULL
area	varchar(255)	Yes	NULL
charge	float	Yes	NULL
v_id	int	Yes	Vendor
address	varchar(1000)	Yes	NULL
phoneno	varchar(30)	Yes	NULL
name	varchar(50)	Yes	NULL
latitude	float(15,12)	Yes	NULL
work_long	float(15,12)	Yes	NULL
work_lat	float(15,12)	Yes	NULL
longitude	float(15,12)	Yes	NULL
accept	tinyint(1)	Yes	NULL