# **Chapter 1: Introduction**

**1.1. Background**

Availability of various online platforms has developed the concept of virtual learning but there are very few school management systems available that would benefit the teachers, students and parents.This project entitled “Kinder” is an online school management system that is accomplished through Django, HTML, JS and CSS.

Every school has a different way of organizing their classrooms but there is merely any system to monitor the activities of small children; basically the ones still in the Kindergartens. So, here we’ve created a centralized platformwhere the activities of those students can be monitored or tracked easily. This would certainly help the parents to know about the current status of their children, keep track of their school activities and progress and also communicate with the teachers.

Similarly, the teachers would also have an easy way to inform the parents about their child’s problems so that the parents can deal with the problems efficiently. Thus, this project is fully focused on managing the schooling of small kids with the help of parents-teachers collaboration.

Several developments have been made in this field regarding keeping track of small kids in Kindergartens. The most recent is the development of the software, Karellen. This service is completely online, and we can access it from any device. The application has a student database, allowing us to easily keep track of basic information such as attendance, meals, sleep, food consumption, and health and safety information. Our website is very similar to this software- as it has a student database, consisting of basic information such as attendance, meals, and classroom activities.

**1.2. Objectives**

**Primary Objectives**

* To provide an easy-to-use interface, robust system to take on the challenge of running an educational system, and the flexibility to adapt to changing needs.
* To keep track of students’ performance, attendance and results.
* To maintain the data of all the students in a database.

**Secondary Objectives**

* To provide an online platform to check upon kids’ performance.
* To help increase parent-teacher collaboration.
* To provide smart routine management, attendance and notices for parents at home.

**1.3.** **Motivation and Significance**

Through reviews and simple research, we came to realize the obvious fact that there are very few number of online school systems that help track the performance of small kids. The available ones are very poorly managed due to the irresponsibility of the school administration resulting either due to the complexity of services or noisy features on the service. As all of the team members are interested in a project to help and make it easier for all parents, teachers, and students, so we choose “Kinder”. Most of the available online school management systems focus on the interaction between students and teachers and they mostly focus on higher grade students. There are not many schools or systems focusing on children and the interaction between parents and teachers. Realizing the disconnect between parents, and teachers in the modern connected world and also how hard it is for parents nowadays to make time to go to school to learn about their child’s school activities and progress and resolve their concerns about their child with their teachers. Since we couldn’t find any platform that does this effectively, we aim to create a platform for parents, students, and teachers to interact with each other easily and make school easier for them all. Through “Kinder” the schools are provided with a platform where teachers can notify parents about the activities taking place in the class, the child’s result and attendance. The parent is also notified about the child’s school routine and the food they are eating during the lunch breaks. Since not many schools provide such a platform, parents have difficulty knowing their child’s school activity and progress so ‘Kinder’ aims to solve this.

# 

# **Chapter 2: Related Works**

**2.1 TCP- The Connect PLus**

TCP (The Connect Plus)[6] is an Educational Institution based software. It is a tool for institute for the real time updates of notices, attendance, diary, homework, assignments, routine, remarks, institute calendar, exam routines, exam syllabus, marksheet, learning materials, events and activities, scheduler, educational news, invitation/greetings, my file, institute information, message from the Head of the Institute to the parents and students who can view it and take benefit of the updates.

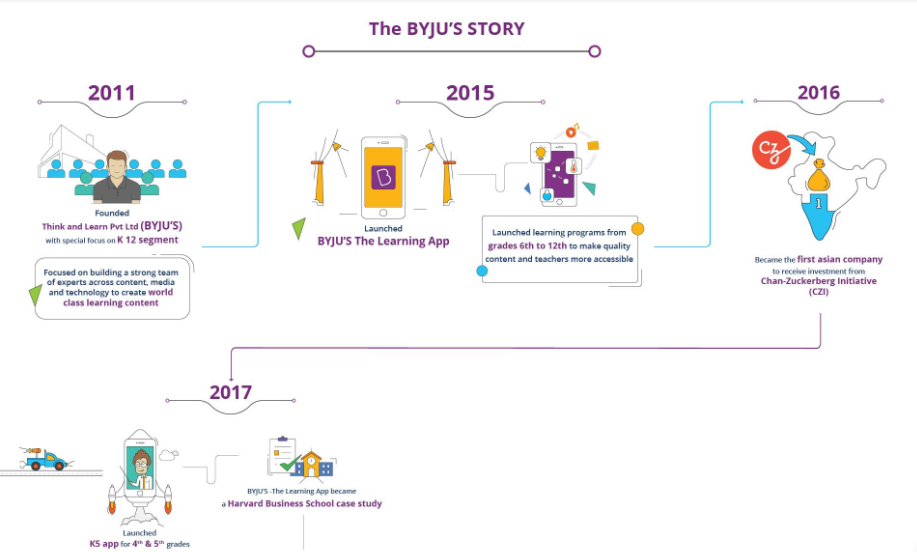
This app mainly focuses on the higher level studies and the management of the curriculum and is merely useful for the parents of small children. So, in order to solve this problem we have developed this interactive site for the sake of the better monitoring of the kids’ daily activities during school by their parents.



**2.2 BYJU’S**

BYJU’S [7] is the world's most valuable ed-tech company and the creator of India's most loved school learning in India. Launched in 2015, BYJU'S offers highly personalised and effective learning programs for classes 1 - 12 (K-12), and aspirants of competitive exams like JEE, CAT, IAS etc. With 35 million registered students and 2.4 million paid subscriptions, BYJU'S has become one of the most preferred education platforms across the globe.

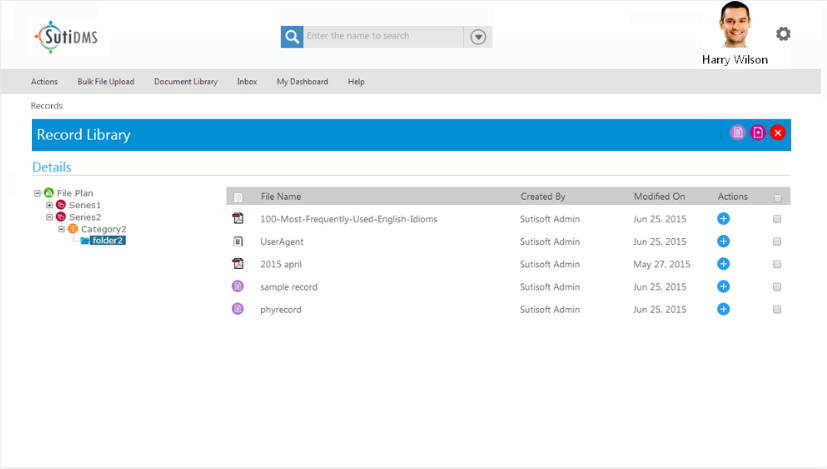
Aspired from this, we’ve developed a simple website that helps to improve the parent-teacher’s collaborative effort for the proper monitoring of the kids (primary level students)



**2.3 SutiDMS**

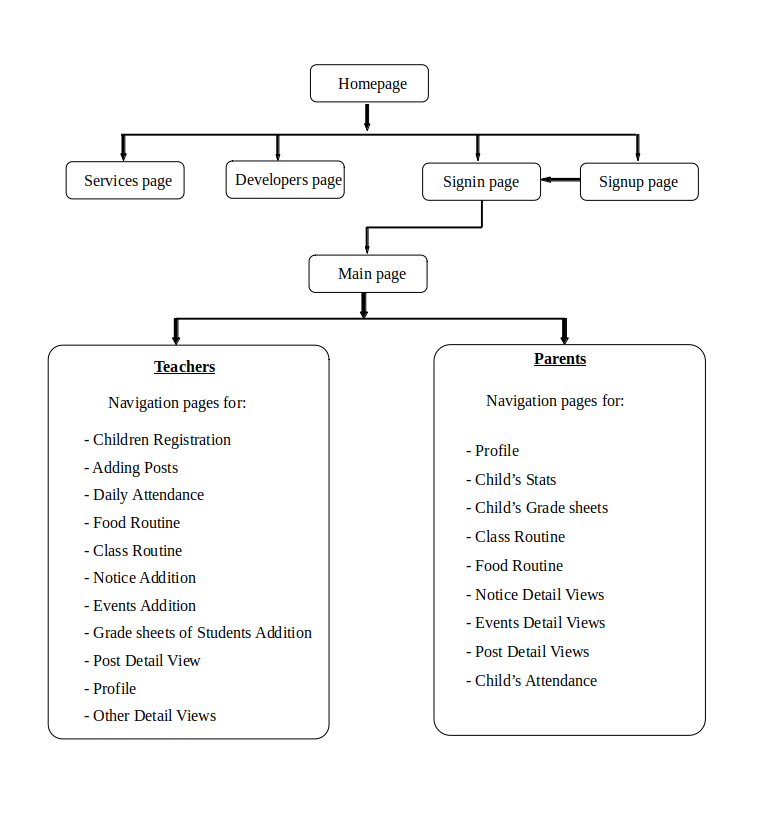
SutiDMS [8] is an online document management software solution that lets users to organize, manage, share and communicate their business-critical information. It comes with integrated modules that help them with team collaboration, workflow management and approvals.

This software is especially based on business work but our site solely focuses on the improvement of the child’s education and development through better guidance. However, the database systems that we’ve used is very similar to the system that this software uses.



# **Chapter 3: Design and Implementation**

**3.1 Website Navigations**



**3.2 System Architecture**

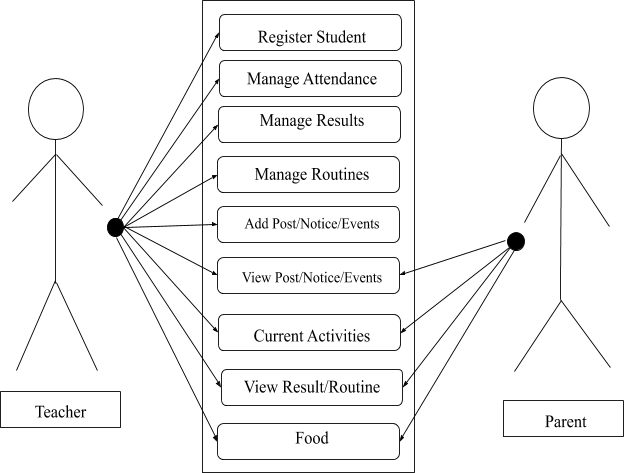
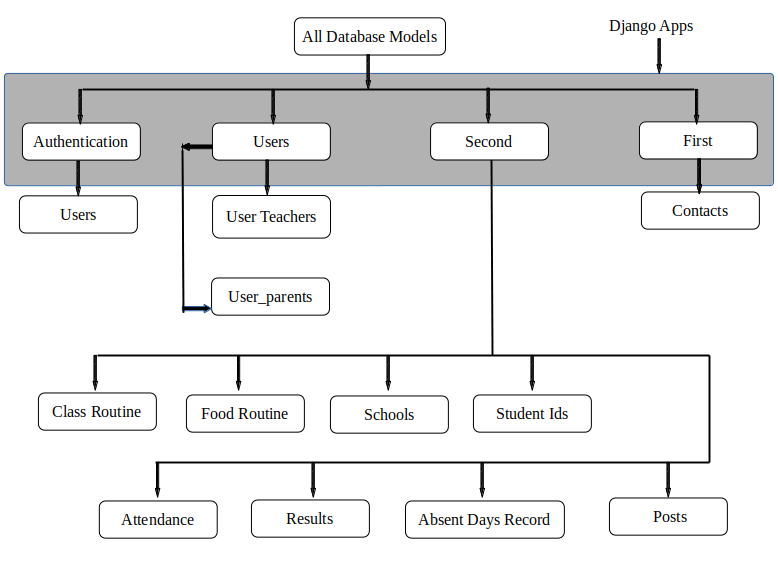


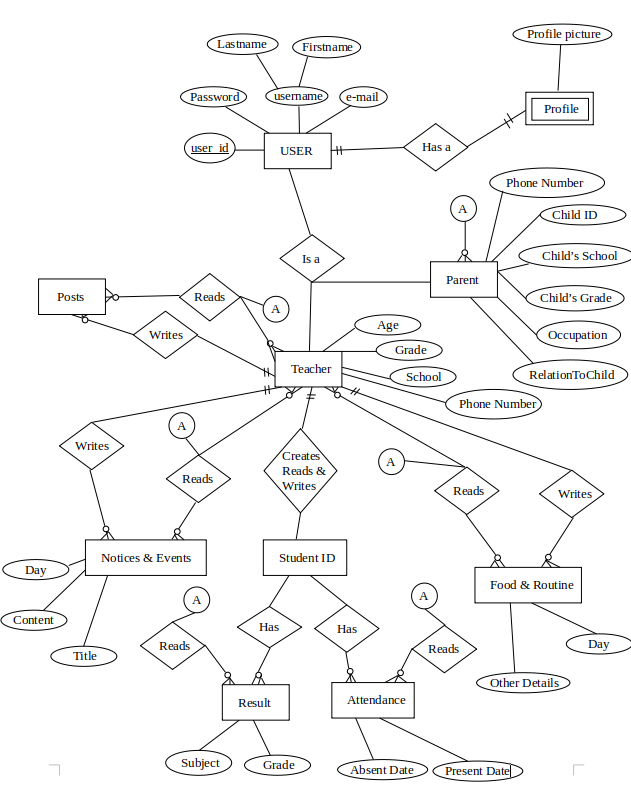
Figure 4: Use Case Diagram

The diagram above is a representation of a user's interaction with the system that shows the relationship between the user and the different features in which the user is involved. Here, the system will manage the activities that would include different calculations, database handling and communications. The teacher will frequently update different sections in the site so that the parents can easily look at the updates from home.

**3.3 All Database Models Used**

****

**3.4 Entity-Relationship Diagram**



**3.5. System Requirement Specification**

**3.5.1. Software Specification**

**3.5.1.1. Front End Tools:**

* **HTML**

Developed by Tim Berners-Lee in 1990; HyperText Markup Language (HTML)[2] is the standard language for documents designed to be displayed in a web browser. It can be assisted with the help of Cascading Style Sheet (CSS) and script languages such as JavaScript.

* **CSS**

Cascading Style Sheets (CSS)[3] is a style sheet language used for describing the presentation of a document written in a markup language like HTML.CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.

* **JS**

Alongside HTML and CSS, JavaScript[5] is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use if for client-side page behavior, and all the major web browsers have a dedicated JS engine to execute it.

**3.5.1.2. Back End Tools:**

* **SQLite DBMS to store/manipulate data:**

SQLite[4] is a relational database management system (RDBMS) contained in a C library. Designed in 2000 by D. Richard Hipp while designing software used for damage-control systems aboard guided missile destroyers.

* **Django Web Framework to develop the website.**

Django[1] is a Python based free and open source web framework, which follows the Model-Template-View (MTV) architectural pattern. Django’s primary goal is to ease the creation of complex, database driven websites.

* **JS**

**3.5.2. Hardware Specification**

* Any device (computer, laptop) with internet access.

**Chapter 4: Discussion on the Achievements**

We started by studying the various languages required for this project through various sources and sites. We also analyzed the features of the systems that we studied and made a clear roadmap for our system flow and how those features were integrated into the system. But translating the roadmaps into code was not as easy as we thought it would be.

We couldn’t decide on what features to add to our platform but after analyzing the platforms similar to our projects available over the internet, we picked the best features which will be the backbone of our system. Through CSS and JavaScript, we improved upon those features and their appearance to make it more pleasing and added new options and layouts as per the requirement to make it easier for parents and teachers to collaborate in our platform.

During sign-up we had to have a unique identification to distinguish between two different users: parents and teachers. Since we had used Django’s default user creation form for sign-up, we couldn’t add additional fields to it. We decided to work around the problem so we tried to use the last name field provided by the Django form as our user identifier but later we decided to connect additional forms in the default user creation form and put our identifiers there. So, during sign-up the user is asked their user type: parent or teacher and is directed to their respective forms automatically.

One of the most exasperating problems encountered was while implementing the attendance management system within our app. At first, we tried handling the attendance repetition problem for a certain day by disabling the buttons, but that later did not prove to be useful because of the reloading of the page. This was probably the most time-consuming task. Later, we solved this by adding simple undo buttons by the side of the present and absent buttons and checking the repetition of the dates in the database. This way we could implement a working attendance system in our webapp.

We couldn't decide on how to authenticate the parent and teacher during sign-up as without authentication anyone could log in to our system. So later we decided on providing each school with a unique school code through which the teachers would be authenticated and a unique child code to individual students through which the parents would be authenticated.

Similarly, all the posts from all the teachers from all schools were shown in the feed of the user so we filtered the feed through school for teachers such that the teachers could see the post from teachers of their school and through grade for parents so parents could see the post of class teachers.

We tried to display student results via Excel Sheets but we were unable to provide parents with individual children’s results. So later we decided to do locally via Django database and JavaScript.

## 

## **4.1 Features**

* **Digital Monitoring:**

With just an access to the internet, parents can digitally monitor their kids via Kinder.

* **Real time Updates:**

Parents will be updated in real time about their kids with the information provided by the teachers.

* **Smart Attendance:**

Attendance of students is taken online by the class teacher and is updated in real-time to the parents.

* **Notice Board:**

There is a notice board panel on the news feed where the class teacher updates notice and the parents are made aware of any upcoming events and holidays.

* **News Feed:**

On the middle of the homepage, there’s a constantly updating list of stories and posts by the class teacher about the kids.

* **Result Section:**

There’s a result section via which the parents can get the progress report of their kids so that they don’t have to bother going to the school to see the results.

**Gantt Chart**

The following Gantt chart represents the time frame for the completion our project by the end of the semester:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ­ **Week**  **Work** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |  |  |
| Research and Study |  |  |  |  |  |  |  |  |  |  |  |  |
| Front End |  |  |  |  |  |  |  |  |  |  |  |  |
| Back End |  |  |  |  |  |  |  |  |  |  |  |  |
| Program Testing |  |  |  |  |  |  |  |  |  |  |  |  |
| Documentation |  |  |  |  |  |  |  |  |  |  |  |  |
| Presentation Preparation |  |  |  |  |  |  |  |  |  |  |  |  |

**Fig 5.1** Gantt chart

# **Chapter 5: Conclusion and Recommendation**

‘Kinder’ was a unique approach creating a school management system for children which we believe helps their teachers and parents interact with each other to create better understanding between them. This project provides the parents and teachers a platform to interact; this keeps the parents updated with the activities that are taking place in their child’s school and also updated with the child’s progress in school and studies.

## **5.1 Limitations**

* Teacher and Parent can’t communicate directly.
* Doesn’t inform parents about the whereabouts of their children.
* Parents can’t give feedback about the webpage.
* Doesn’t notify parents about the update.

## **5.2 Future Enhancements**

* Kinder Mobile App
* Real time communication between parents and teachers.
* Notifications about updates and new notices.
* Geolocation of the child during travel and outing events.

# **References**

[1]Tutorialspoint, (2019, 11 11). *Tutorials-Django.* Retrieved from:

<https://www.tutorialspoint.com/django/index.htm>

[2] W3C. (2019, 11 11). *Learn HTML*. Retrieved from W3Schools:

<https://www.w3schools.com/html/default.asp>

[3] W3C. (2019, 11 11). *Learn CSS*. Retrieved from W3Schools:

<https://www.w3schools.com/css/default.asp>

[4] Tutorialspoint, (2019, 11 11). *Tutorials-SQLite.* Retrieved from:

[https://www.tutorialspoint.com/sqlite/index.htm](https://www.tutorialspoint.com/mysql/index.htm)

[5] W3C. (2019, 11 11). *Learn JS*. Retrieved from W3Schools:

<https://www.w3schools.com/js/default.asp>

[6] The Connect Plus TCP - Apps on Google Play. (n.d.). Retrieved from:

<https://play.google.com/store/apps/details?id=com.tcp.theconnectplus&hl=e>

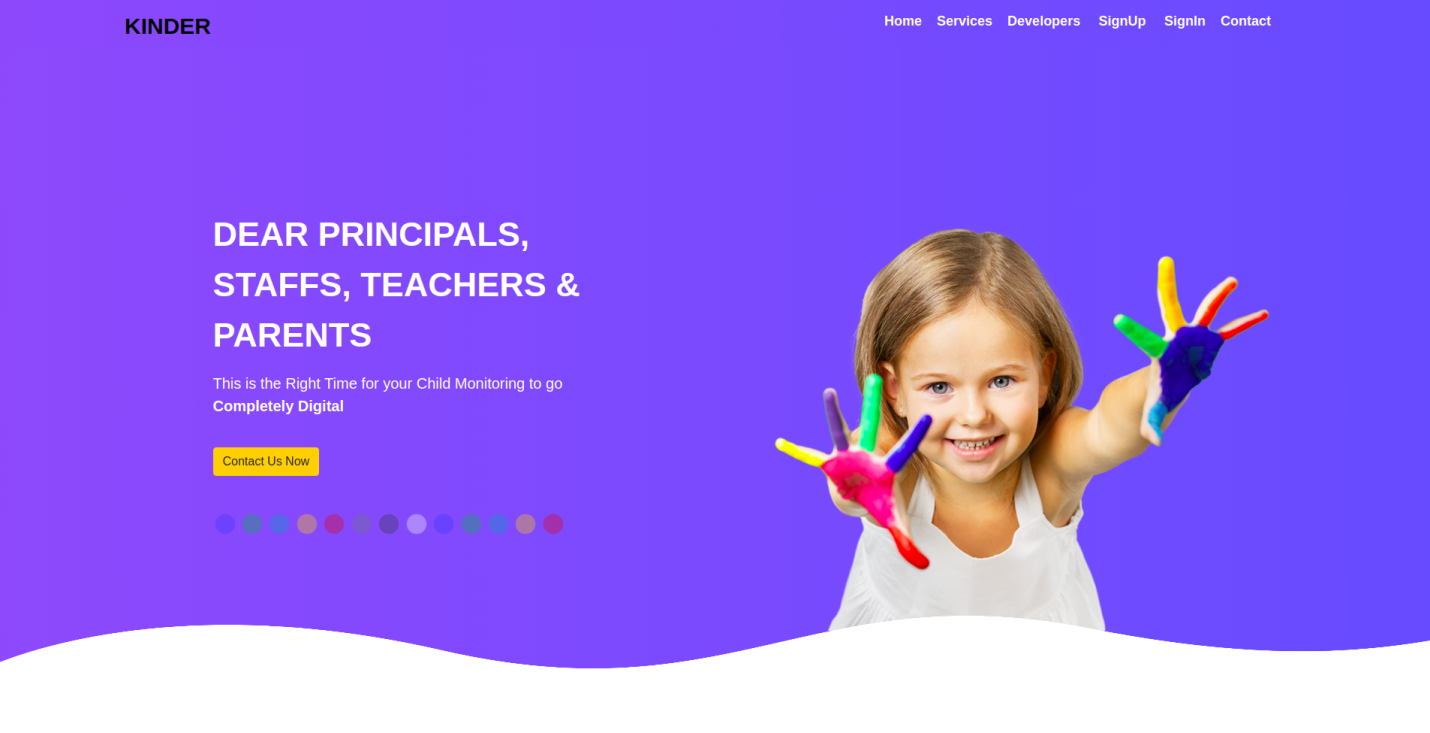
[7] BIJU’S - Apps on Google Play. (n.d.). Retrieved from:

<https://play.google.com/store/apps/details?id=com.byjus.thelearningapp&hlen>

[8] SutiDMS - Web Application. (n.d.). Retrieved from:

https://www.sutisoft.com/sutidms/

# **APPENDIX**



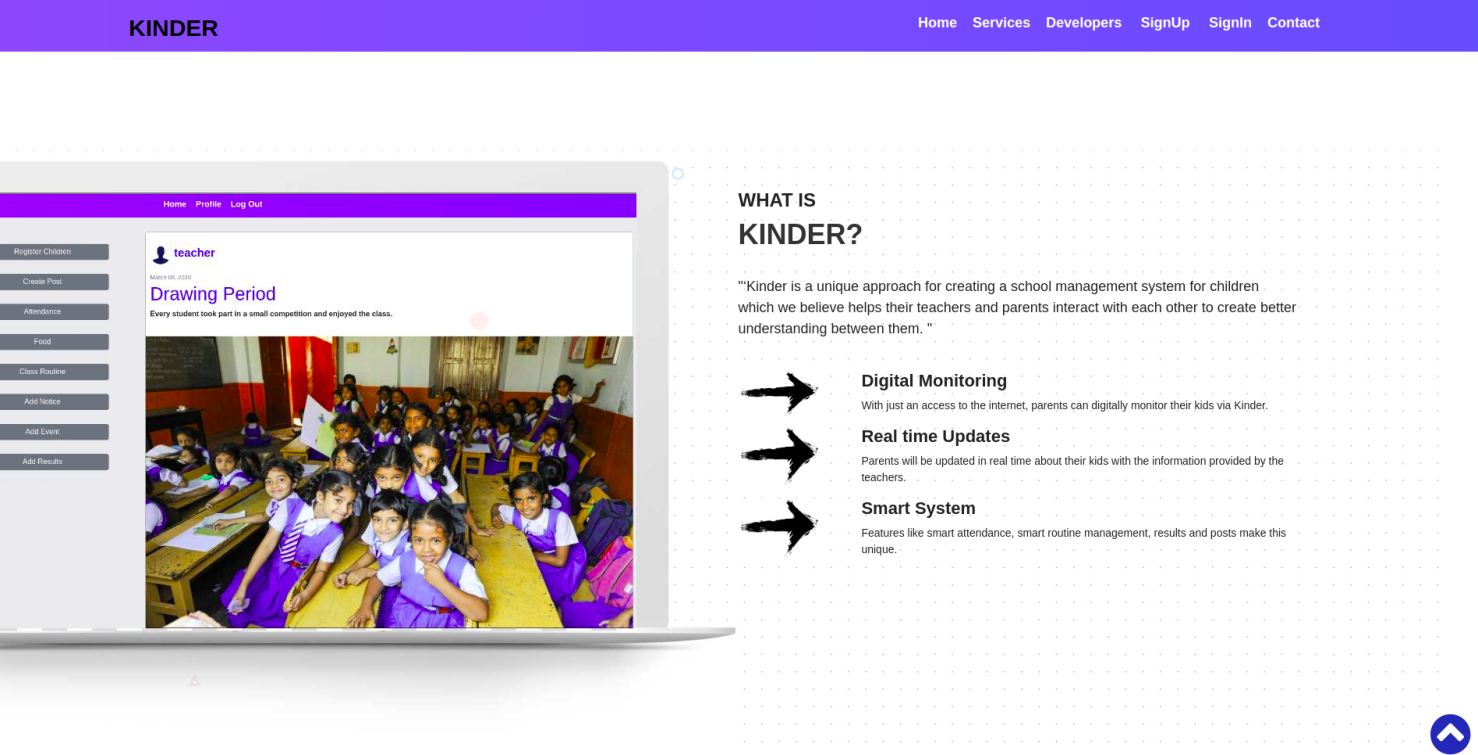


Fig: Homepage

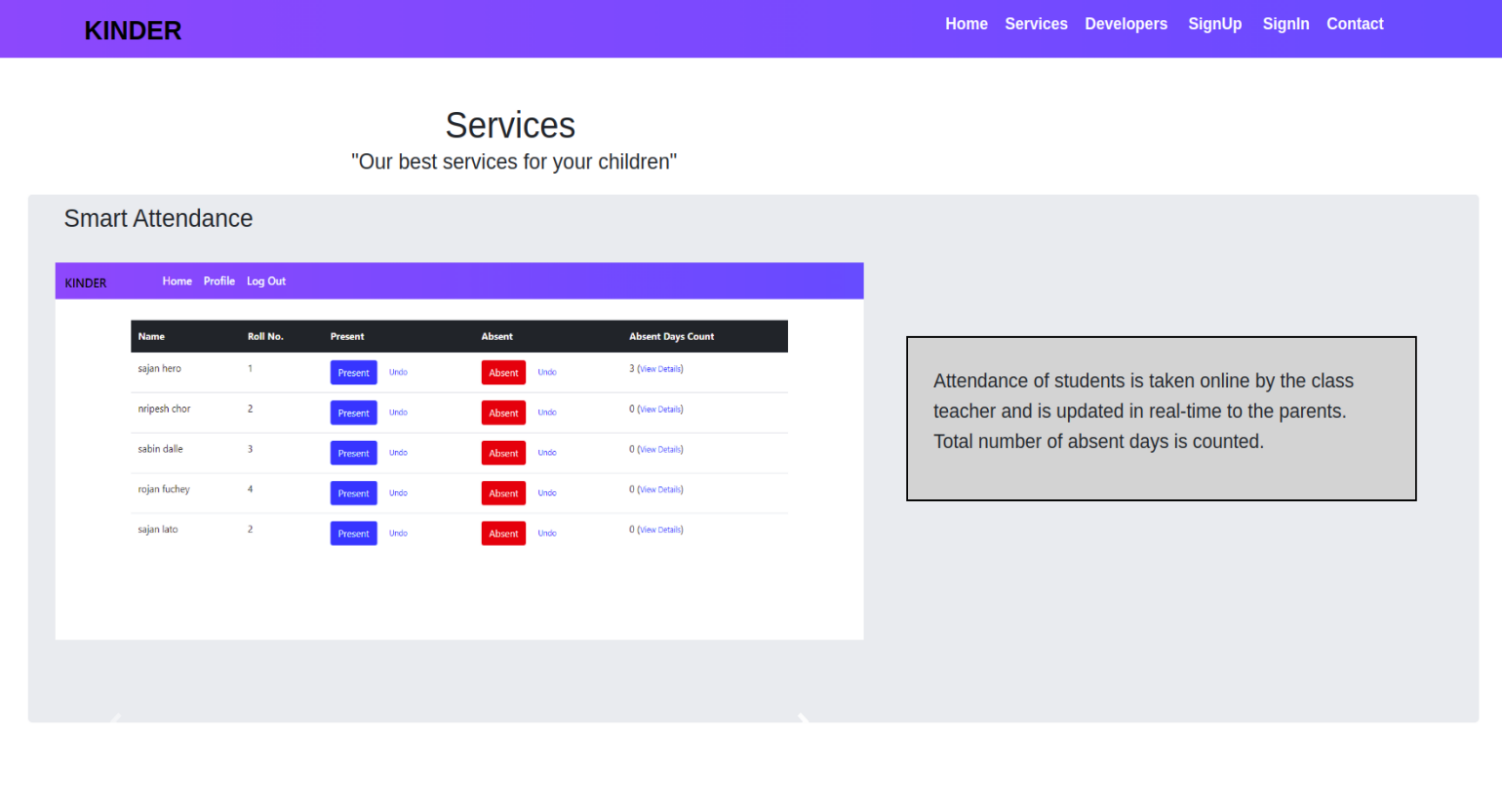


Fig: Services Page

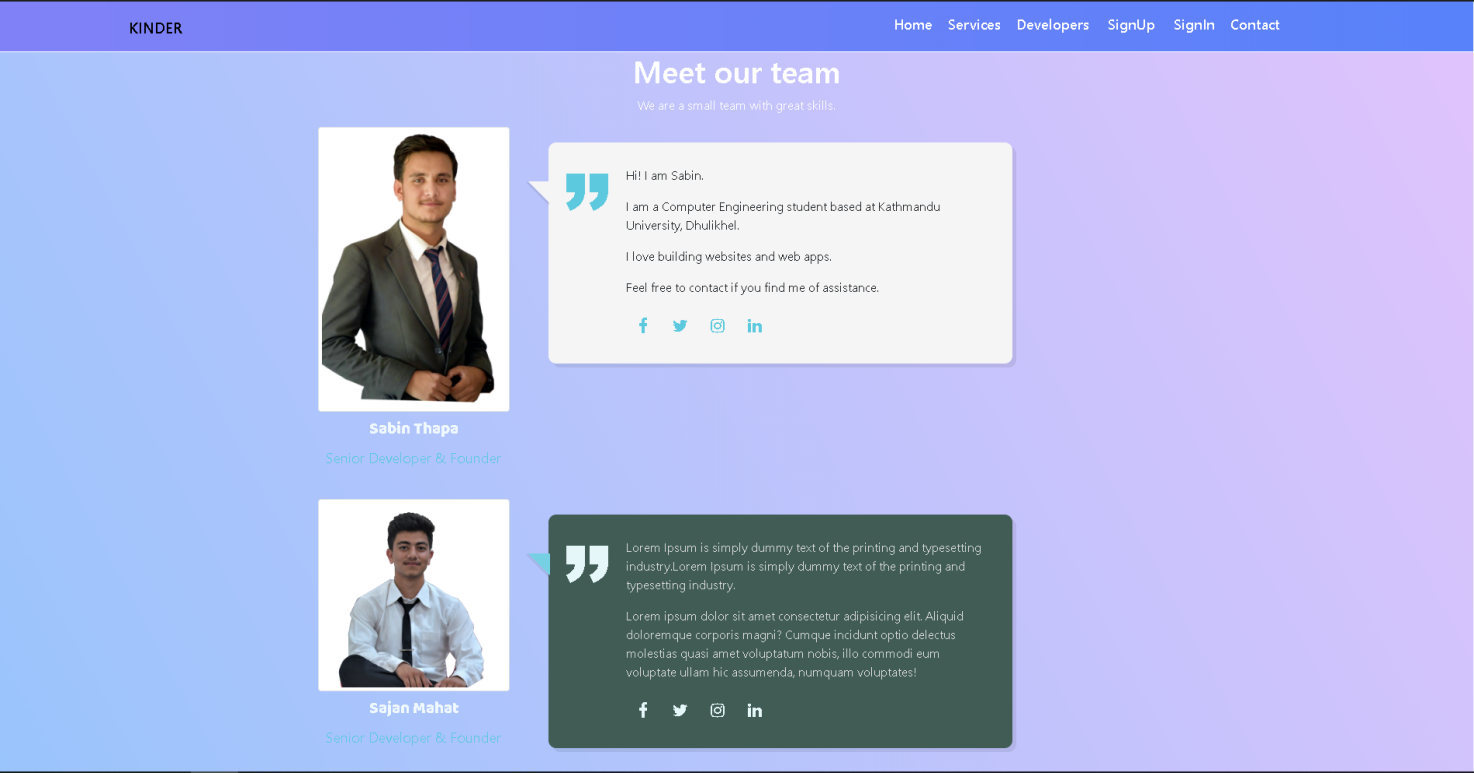


Fig: Developers Page

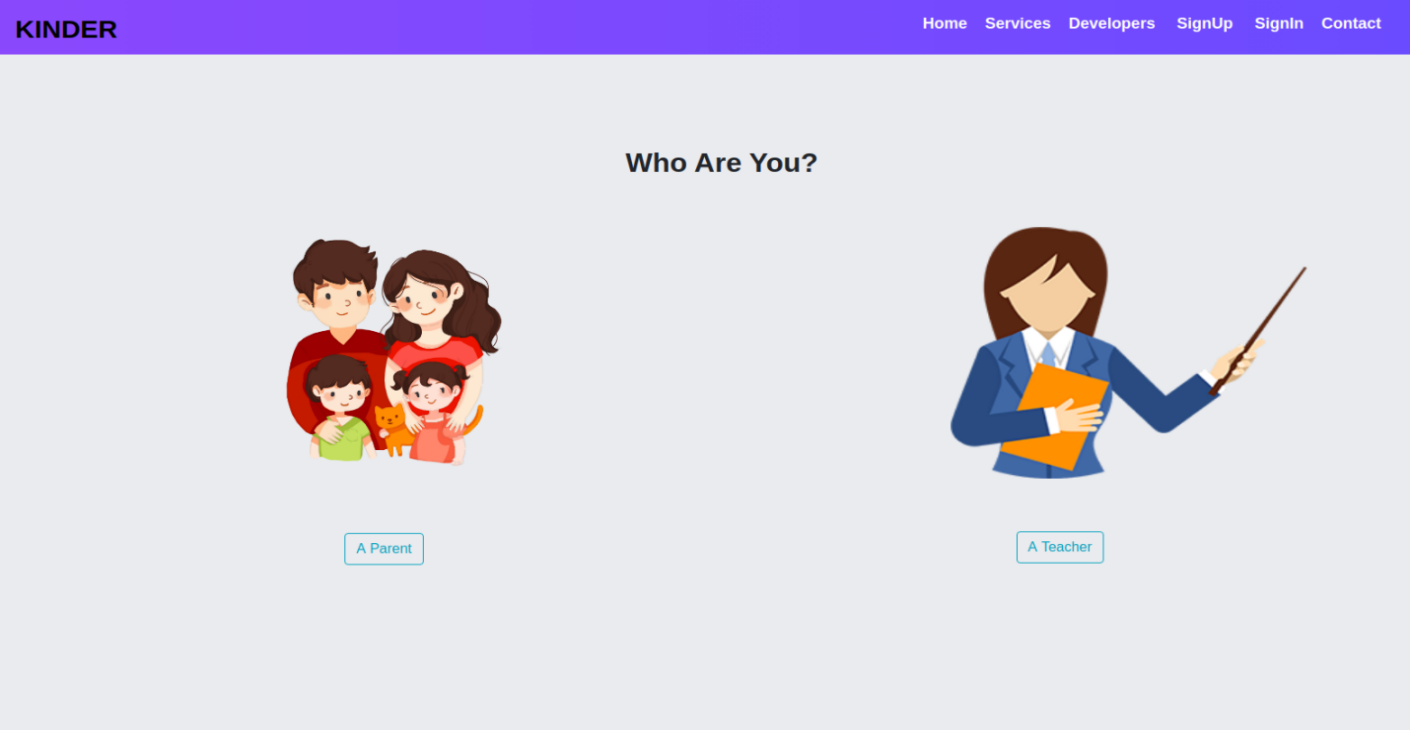


Fig: Main SignUp page

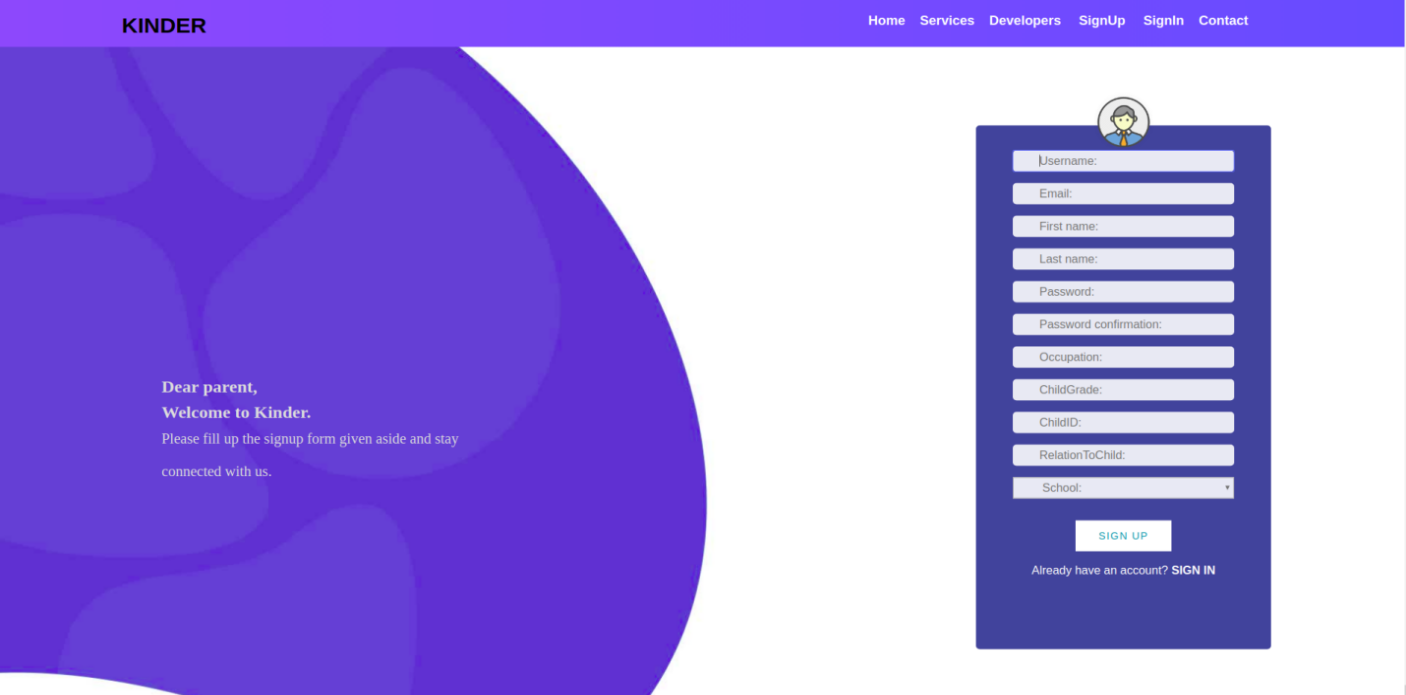


Fig: Parent’s Signup Page

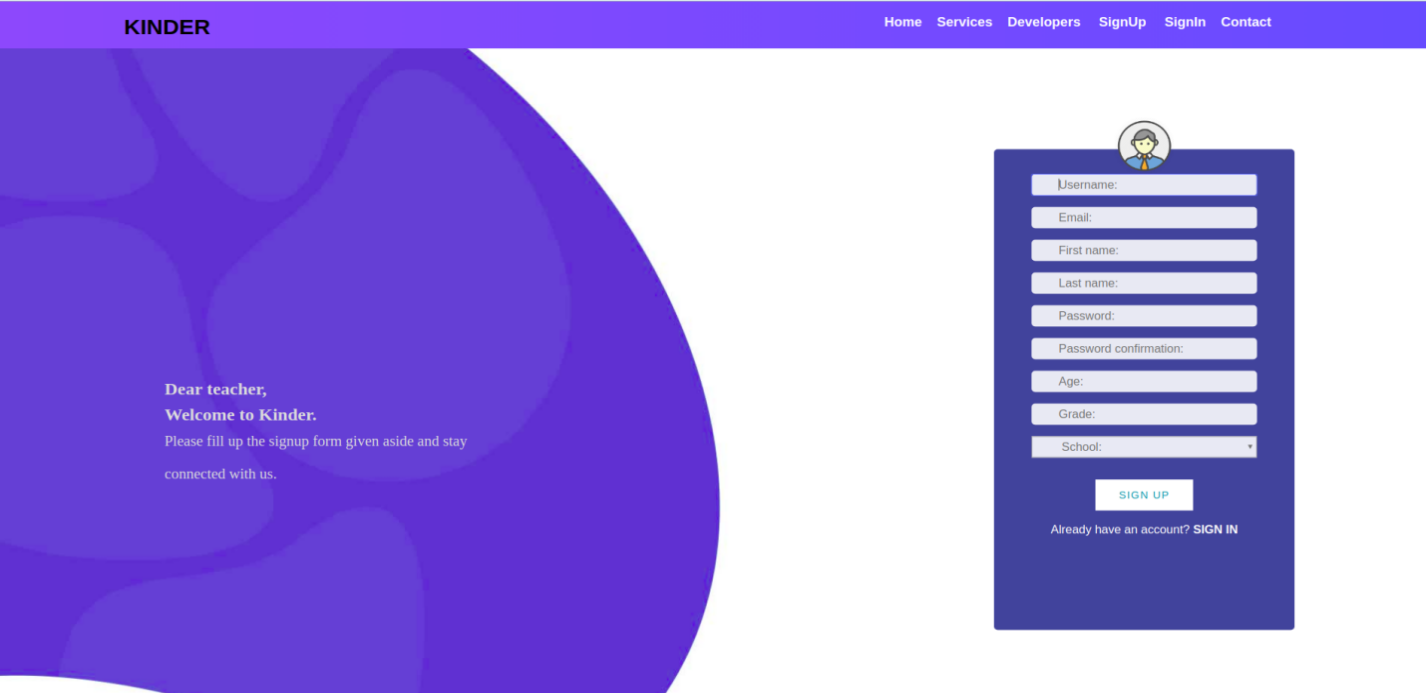


Fig: Teacher’s Signup Page

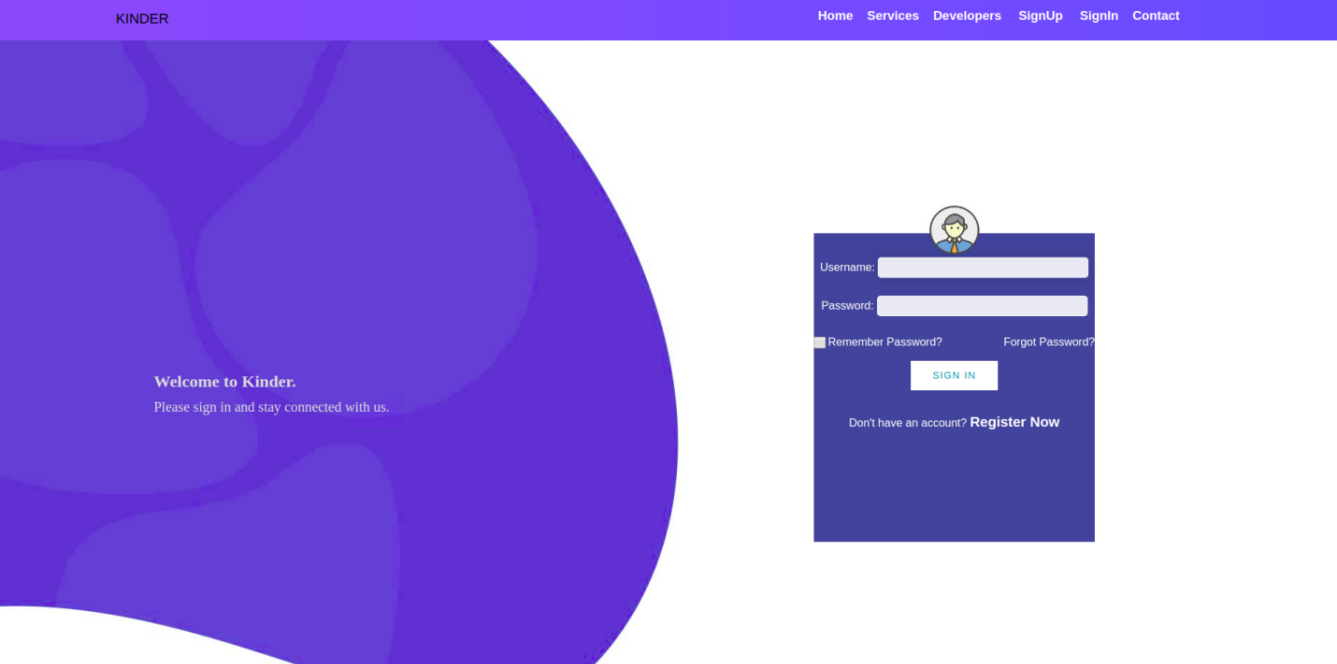


Fig: Login Page

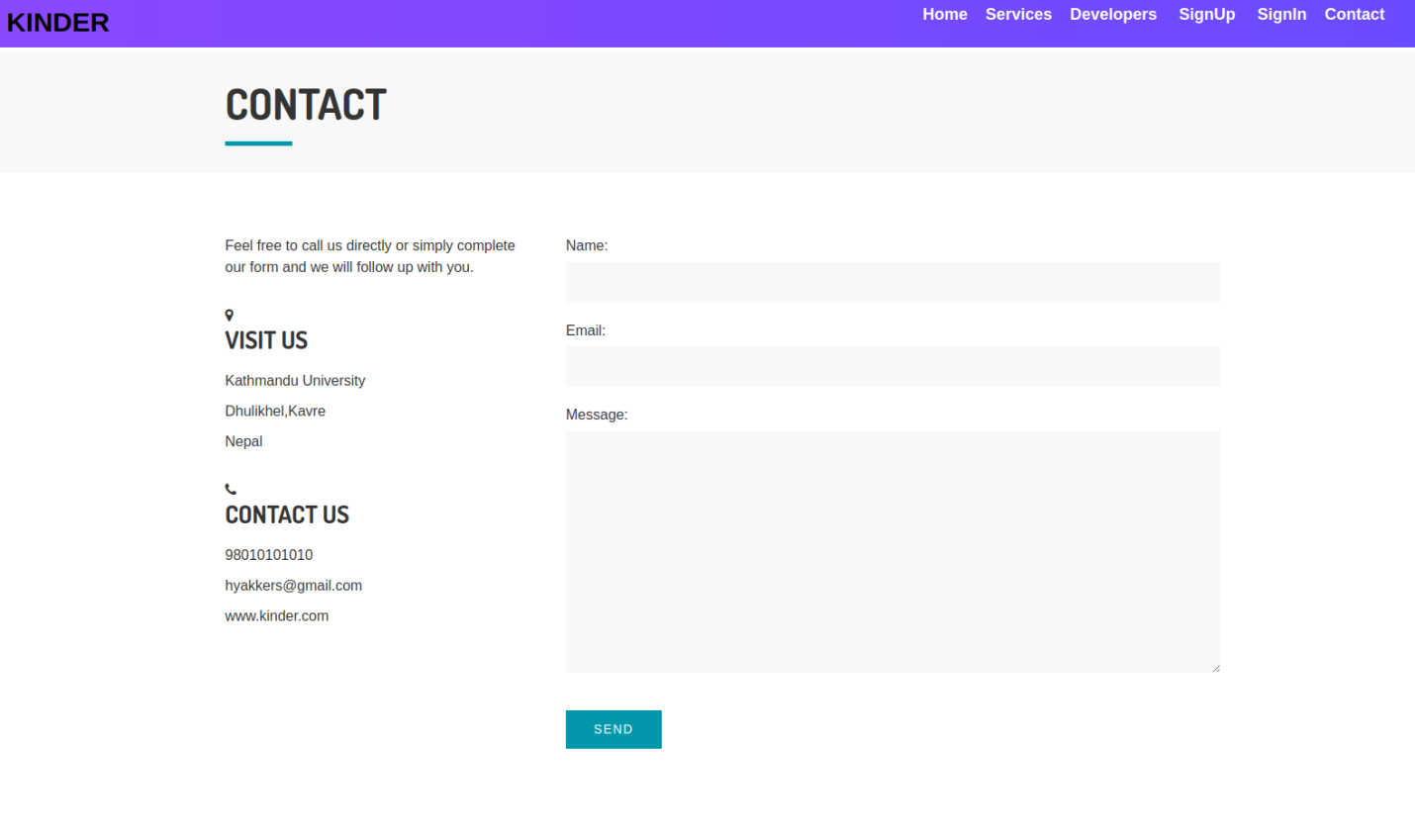


Fig: Contact Page

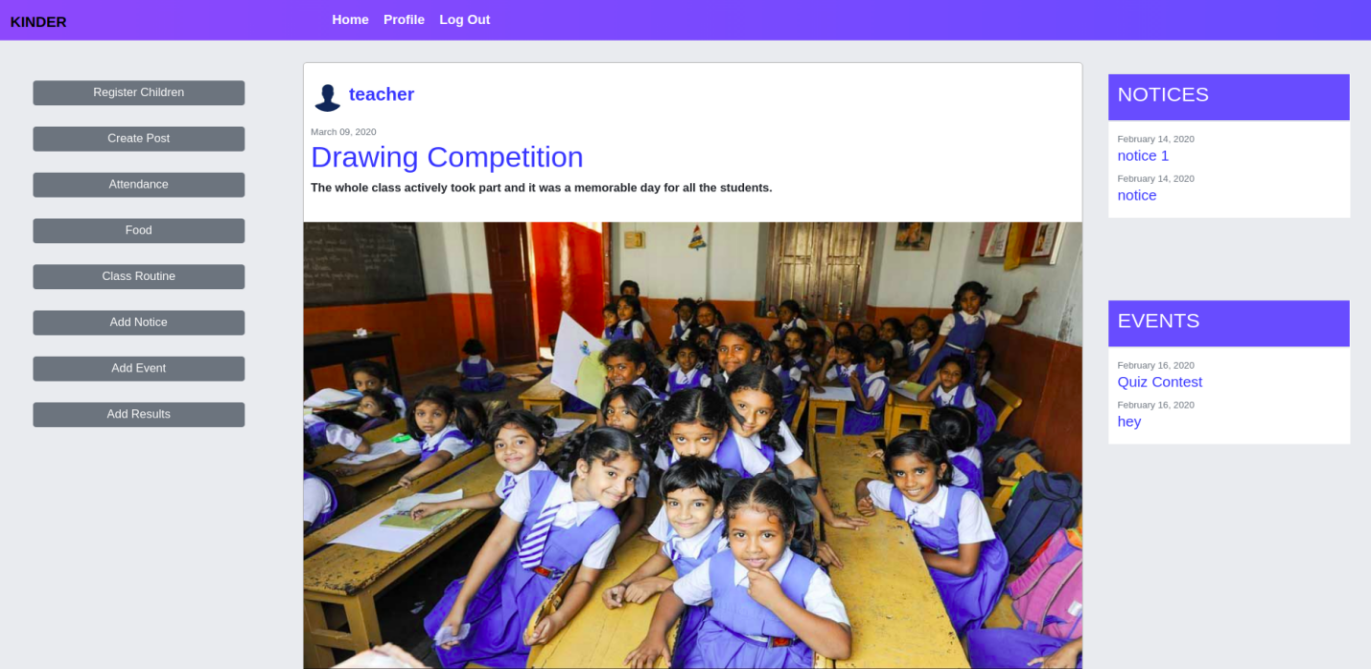


Fig: Teacher’s Feed

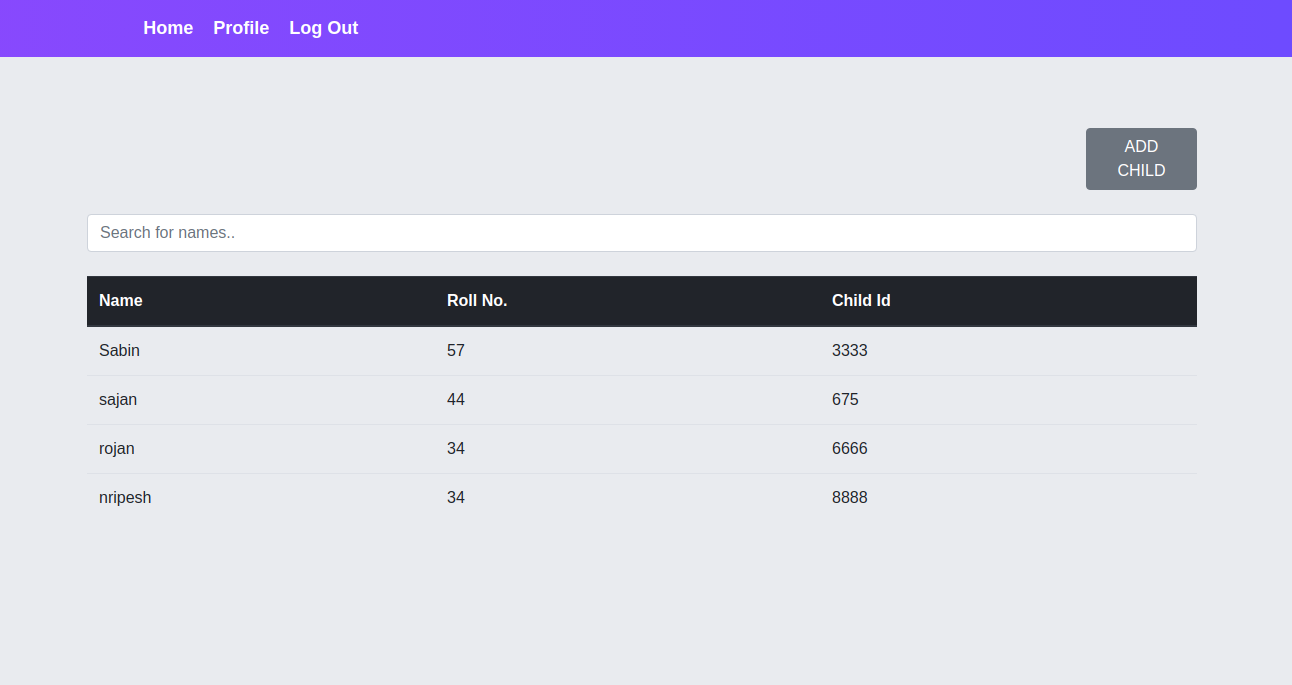


Fig: Child Registration Page

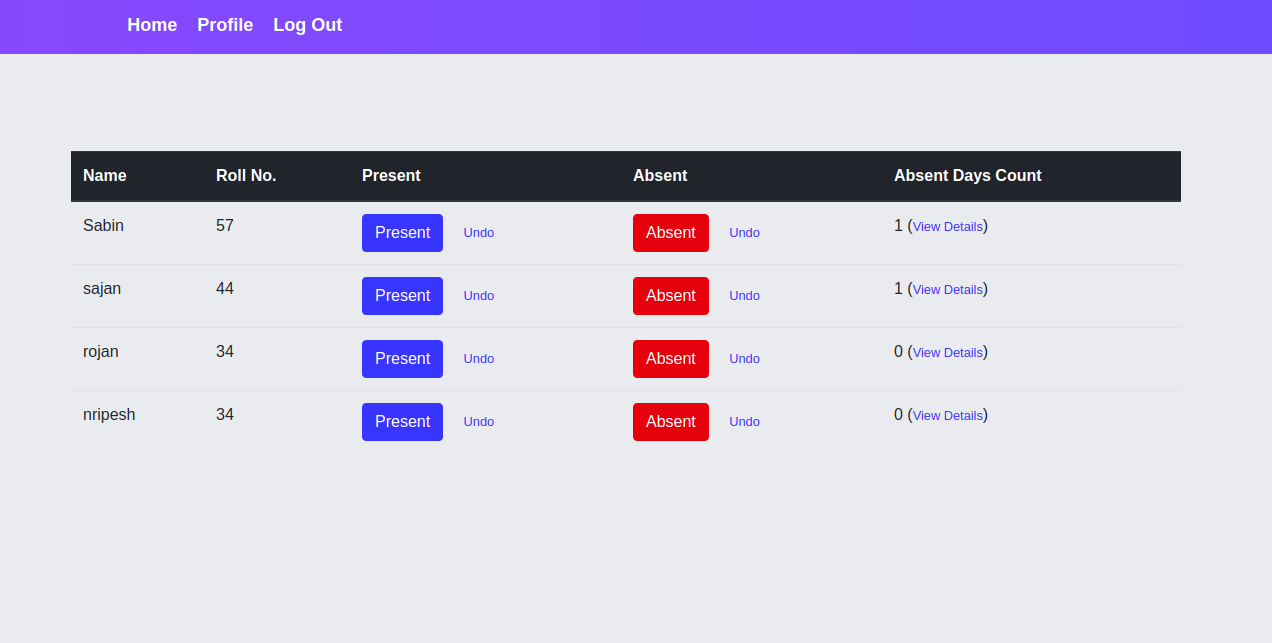


Fig: Attendance Page

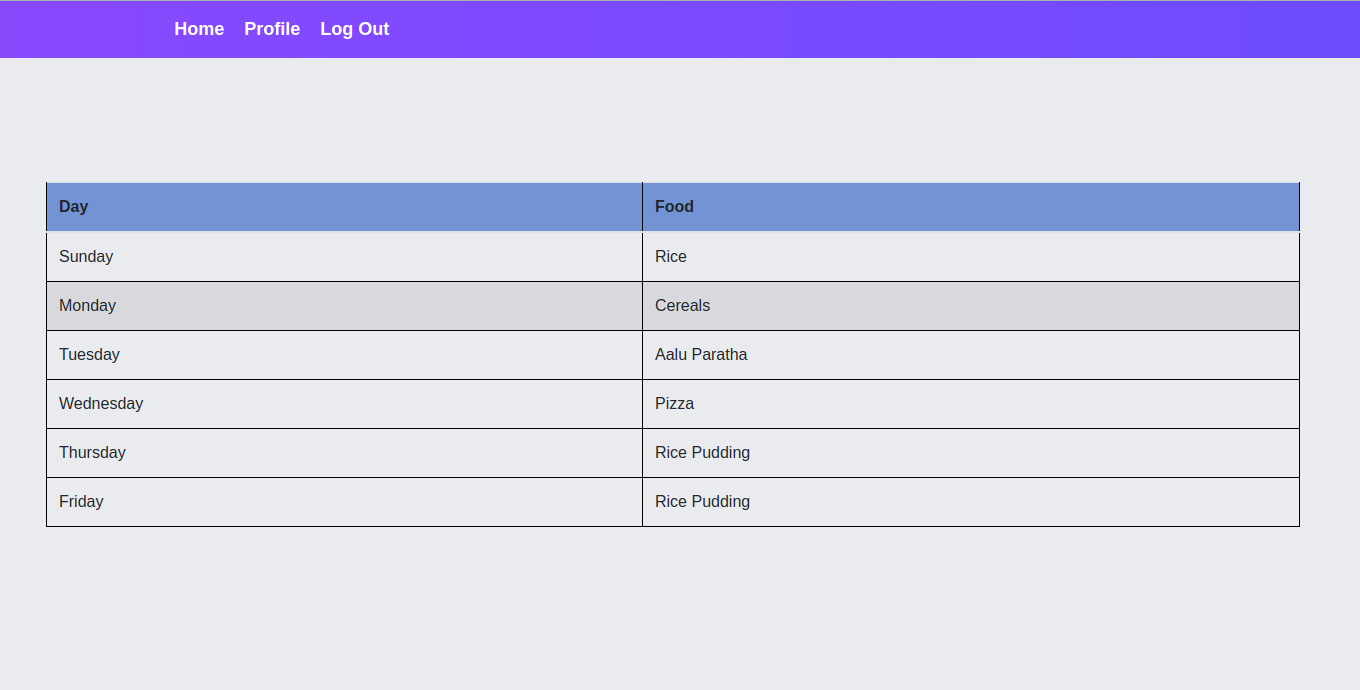


Fig: Food Routine

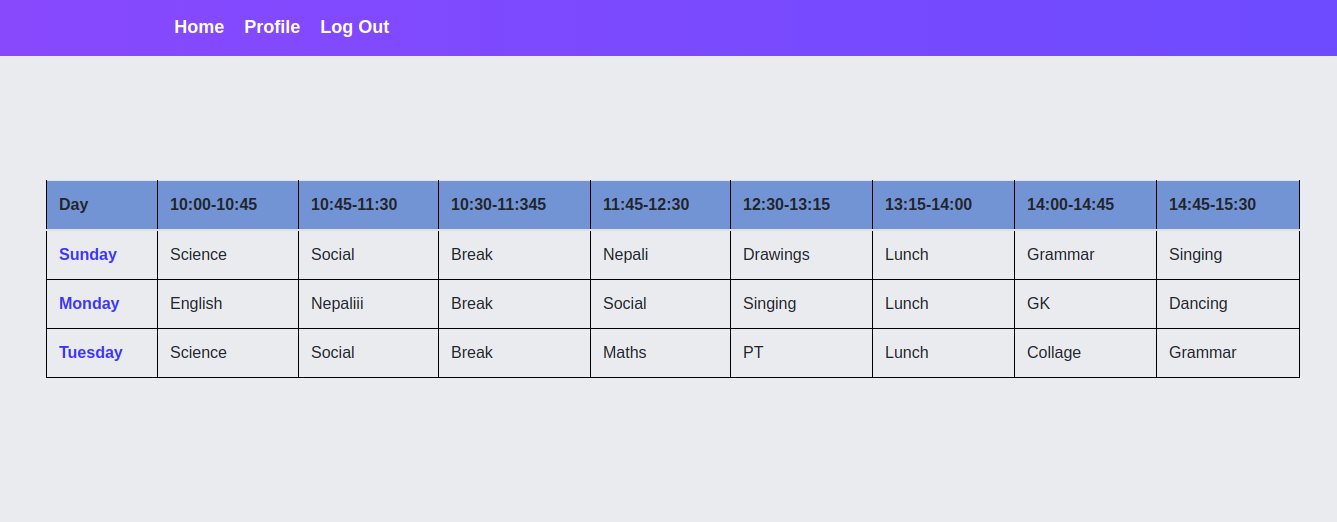


Fig: Class Routine

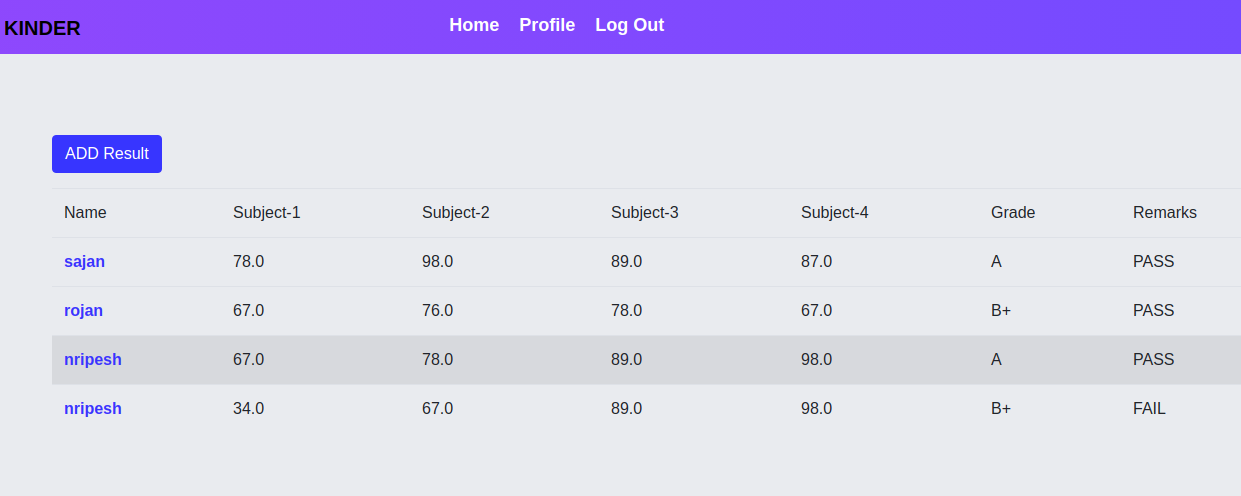


Fig: Results Section

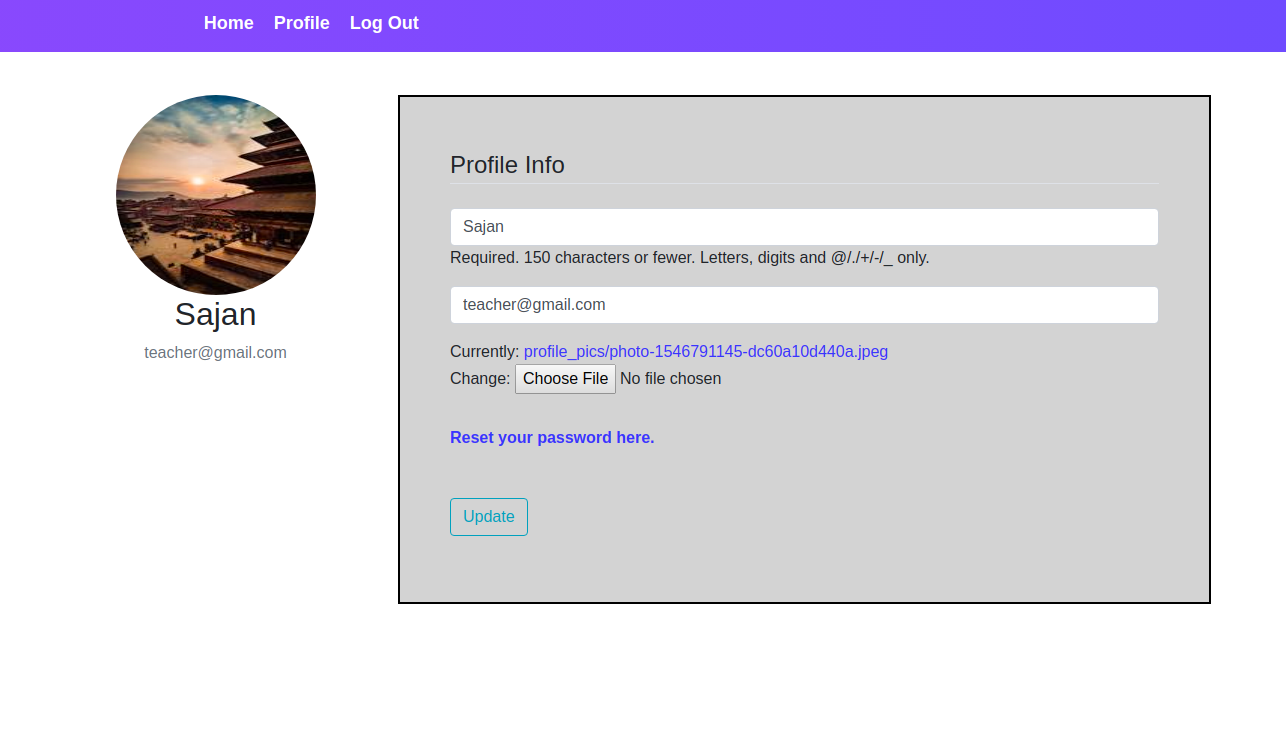
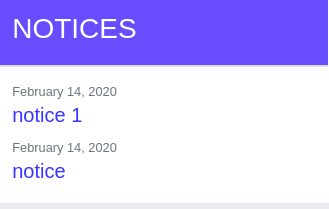


Fig: Teacher’s Profile



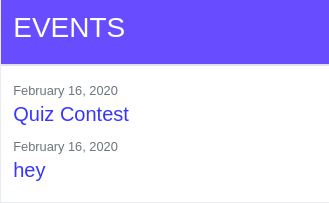


Fig: Notices and Events Panel