



# Resume Builder

(A Web Application)

# Team-01

- Ravi Krishna Teja Dachepalli
- Jayanth Golla
- Manoj Desetty
- Anudeep Tippireddy



# Background Story

- Individuals finding it difficult to create a Resume from scratch , in any job application process or in any other business. It must be kept short, straightforward, and current with work experience, and it must be updated on a regular basis.
- One of my friends found it hard to build and modify a resume every single time while applying for different on campus jobs. So , this made us think about an online web application that will automatically generate a good format resume from the input entered by the user.

# Data

- Here the data is Users Input. Users input is taken as raw data and this data will be formatted to give a resume.
- This project doesn't need any sampling as the user gives only the requested data.
- This data will be fresh for every user and this application will not be working with any global or previous data.
- This data doesn't contain geographical information or any other factors like regional, national or global as the data is user specific.

# Main Page

- Main screen, where user can fill out the details for our resume.
- This data will be used to build resume in different templates according to the user selection.

Ry Resume

Profile

HEADING

Profile

PHOTOGRAPH

FIRST NAME

Jayanth

LAST NAME

Golla

SUBTITLE

Student

DATE OF BIRTH

10/31/2021

ADDRESS LINE 1

525 E Armour Blvd, Apt# 605

ADDRESS LINE 2

Apt# 605

CITY

Kansas City

ZIP CODE

64109

Jayanth Golla

Student

Profile

Address

525 E Armour Blvd, Apt# 605

Apt# 605

Kansas City 64109

Phone Number

+917013878283

Email Address

golla.jayanth09@gmail.com

Date Of Birth

31 October 2021

OBJECTIVE

I want to work as a Software Developer.

WORK EXPERIENCE

Sukshi

Full Stack

(December 2020 - November 2021)

Worked on Video Conferencing app.

EDUCATION

UMKC

Masters Computer Science

(January 2021 - Present)

Templates

Onyx

Pikachu

Gengar

Castform



# Features

- User Authentication

User can create an account or sign in through Gmail or sign in anonymously

**Rx**  
**Resume**

## Reactive Resume

A free and open source resume builder.

Login

Who are you?

X

Reactive Resume needs to know who you are so it can securely authenticate you into the app and show you only your information. Once you are in, you can start building your resume, editing it to add new skills or sharing it with the world!

Cancel

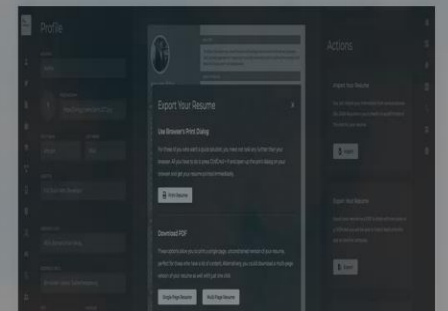
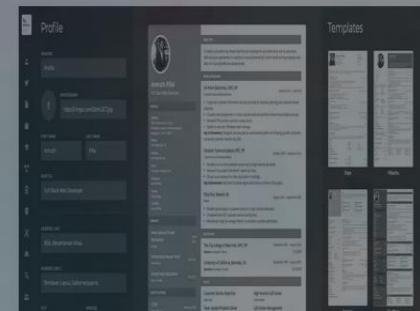
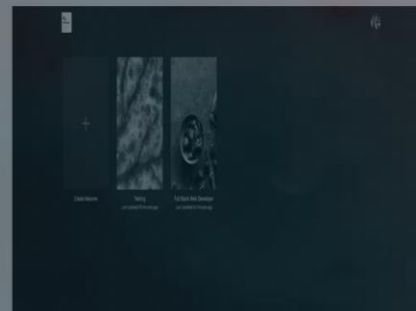
Sign in with Google

Sign in Anonymously

Reactive Resume is a free and open source resume builder. You can create multiple resumes, share them with recruiters through a unique link.

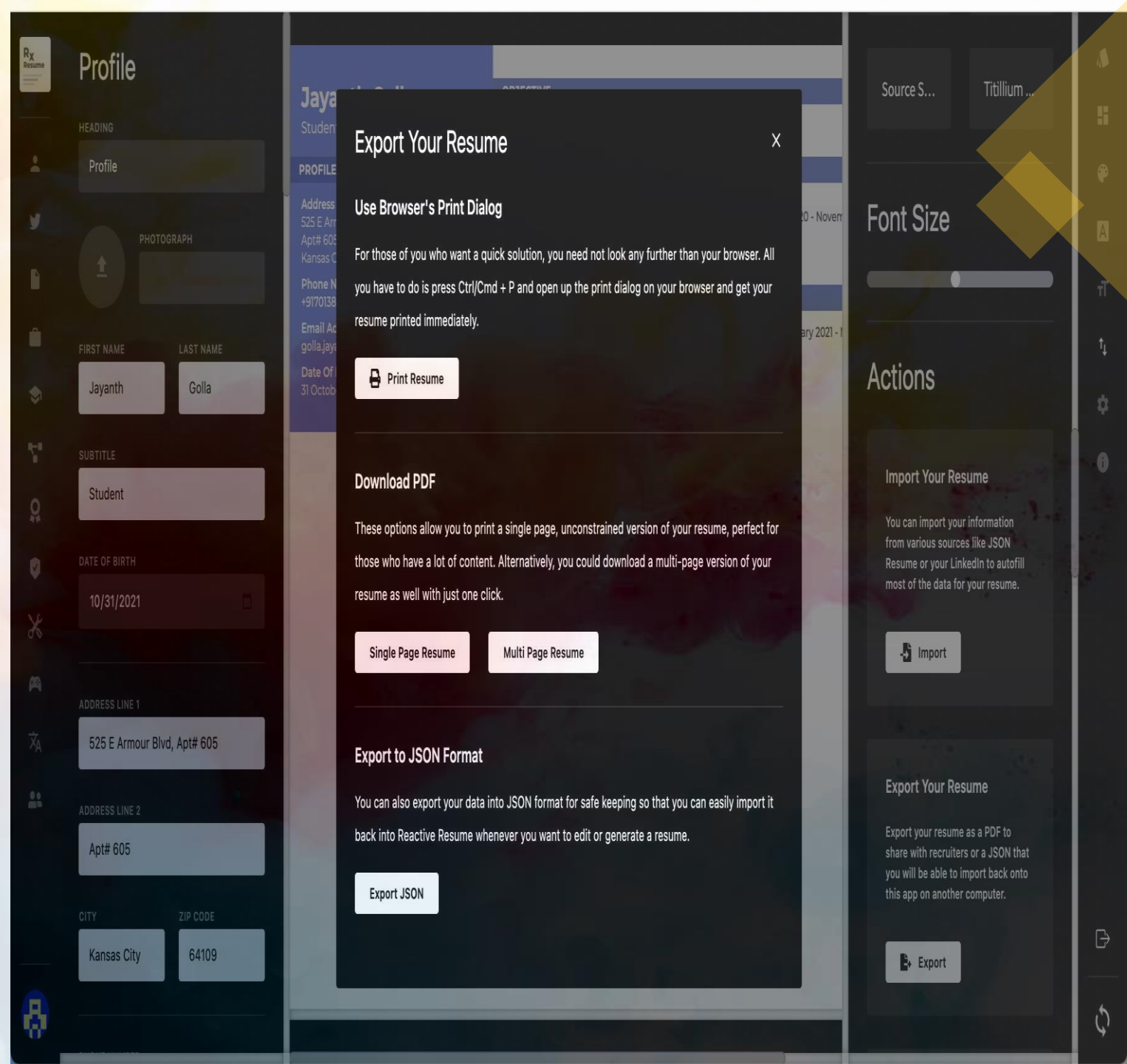
as 1, 2, 3. With this app, you can create multiple resumes, share them with recruiters through a unique link.

## SCREENSHOTS



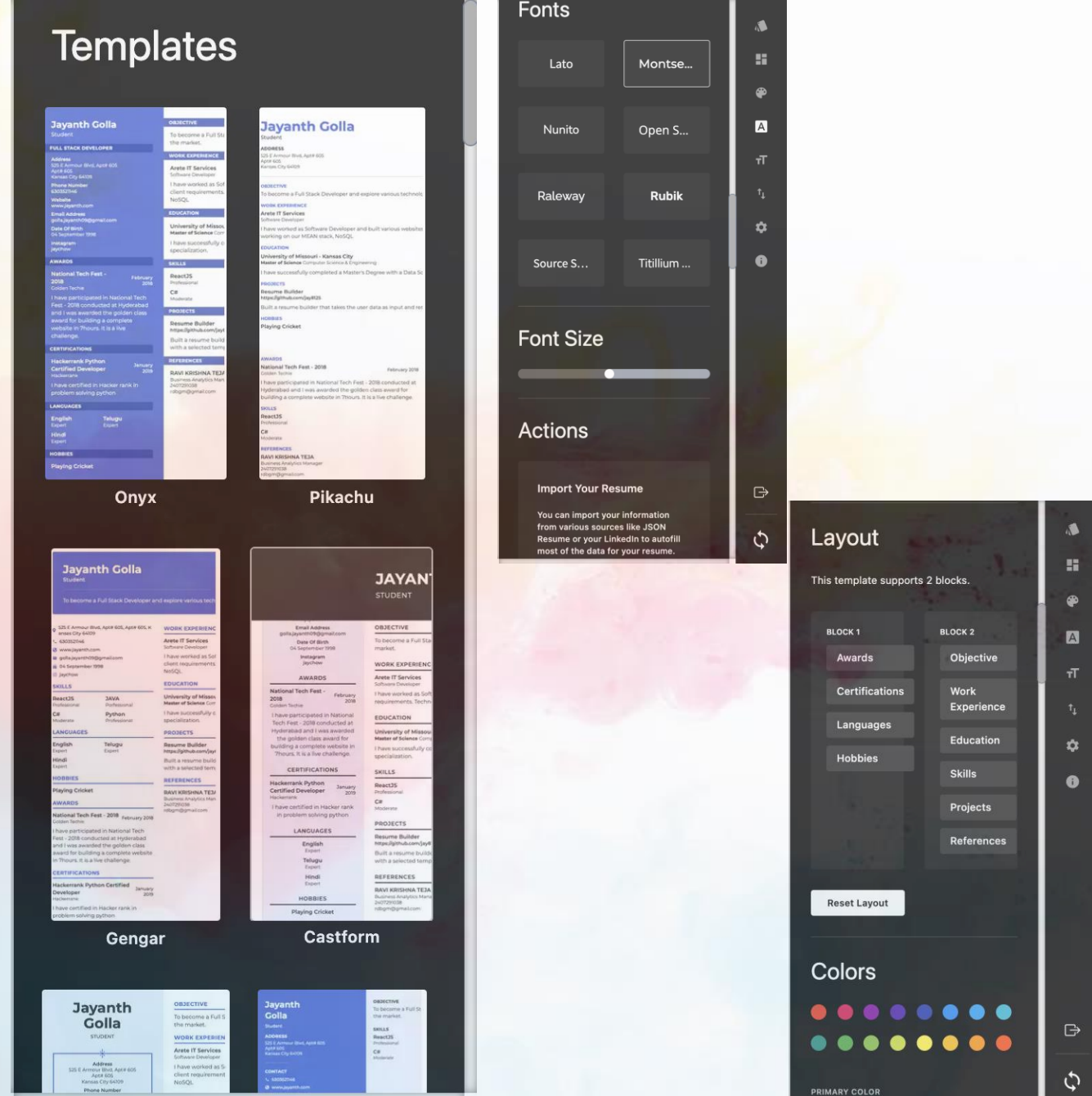
# Features

- Another important feature is that user can import, export, print and download resume.
- This enables user friendly experience.



# Features

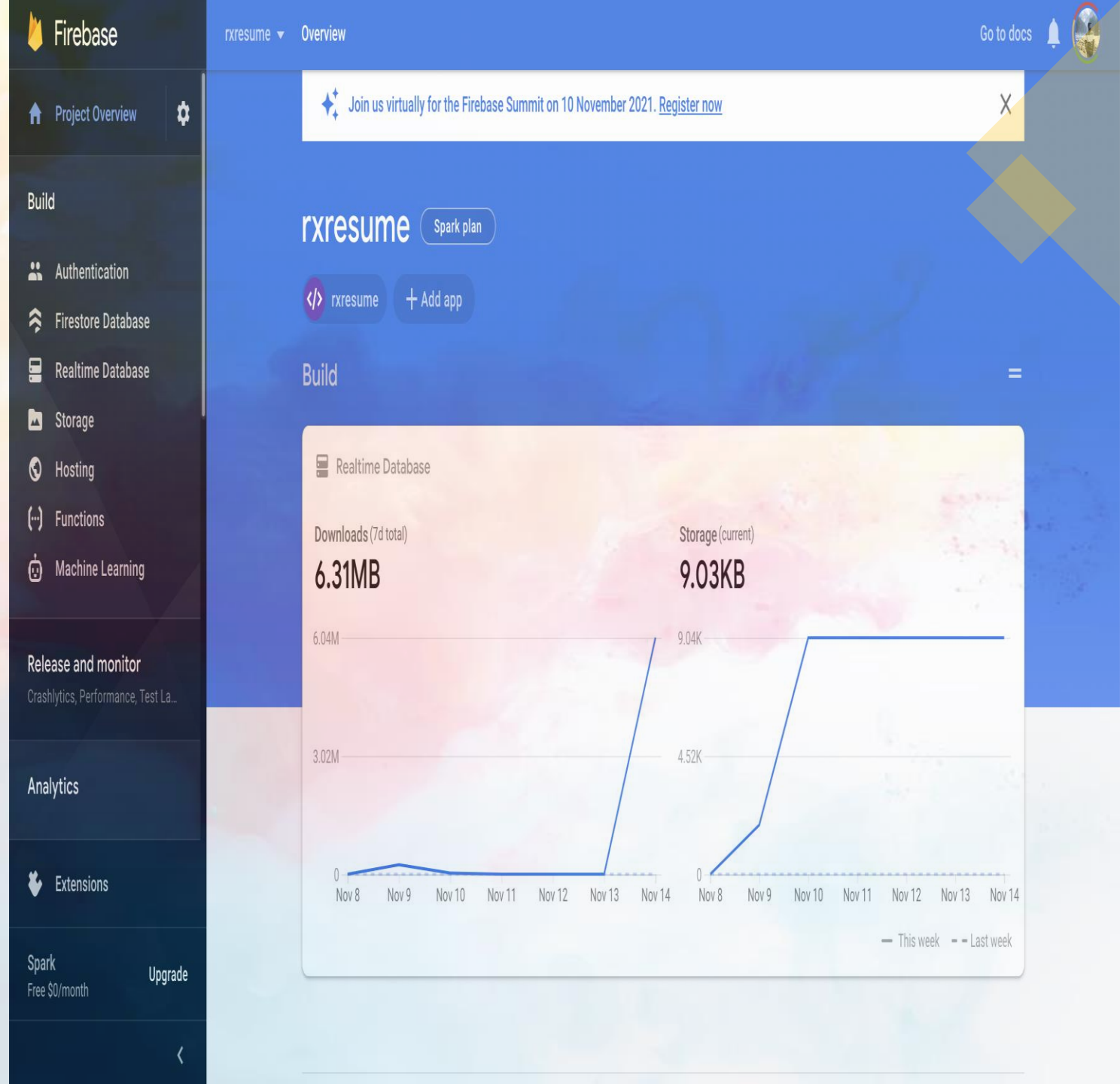
- User can select any template out of the 6 provided templates.
- User can change the color of the template.
- User can access font, font size, layout colors etc. in the side bar provided.





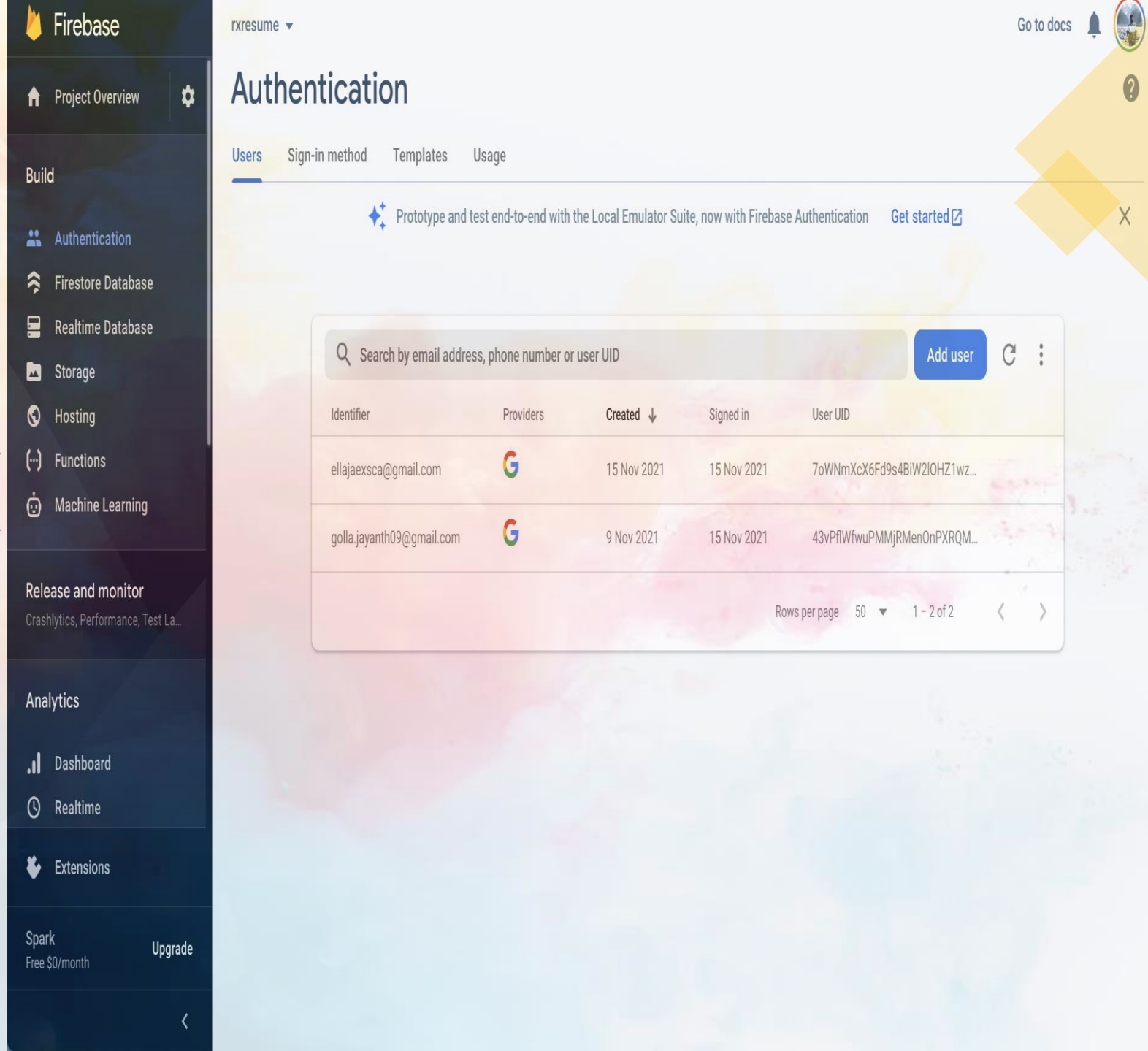
# Database

Firebase Realtime Database is used to store the data and database schema is created according to the requirement.



# Database

- For Authentication, you can login through Gmail or Anonymously.
- The picture displays the list of users in the database.



The screenshot shows the Firebase Authentication console. The left sidebar contains the following menu items: Project Overview, Build (Authentication, Firestore Database, Realtime Database, Storage, Hosting, Functions, Machine Learning), Release and monitor (Crashlytics, Performance, Test Lab), Analytics (Dashboard, Realtime), Extensions, and Spark (Free \$0/month, Upgrade). The main content area is titled 'Authentication' and has tabs for Users, Sign-in method, Templates, and Usage. The 'Users' tab is selected, displaying a table of users. At the top of the Users tab, there is a search bar and an 'Add user' button. Below the search bar is a table with the following columns: Identifier, Providers, Created, Signed in, and User UID. The table contains two rows of user data. At the bottom right of the table, there is a pagination control showing 'Rows per page 50' and '1 - 2 of 2'.

Identifier	Providers	Created	Signed in	User UID
ellajaexsca@gmail.com	G	15 Nov 2021	15 Nov 2021	7oWNmXcX6Fd9s4BiW2IOHZ1wz...
golla.jayanth09@gmail.com	G	9 Nov 2021	15 Nov 2021	43vPfiWfwuPMMjRMenOnPXRQM...

# Database

- This picture displays the space to write schema for the database.

The screenshot displays the Firebase Realtime Database interface. On the left is a dark sidebar with the 'Realtime Database' option selected. The main panel shows the 'Data' tab for the database 'rxresume-6f067-default-rtdb'. The URL bar indicates the database's location. The schema is visualized as a tree structure with the following nodes: awards, category, certifications, education, hobbies, languages, metadata, objective, projects, public: true, references, resumes, skills, and social. Each node is represented by a green box with a plus icon to its left.

rxresume ▾

## Realtime Database

Data Rules Backups Usage

Protect your Realtime Database resources from abuse, such as billing fraud or phishing [Configure App Check](#)

<https://rxresume-6f067-default-rtdb.firebaseio.com/>

rxresume-6f067-default-rtdb

- awards
- category
- certifications
- education
- hobbies
- languages
- metadata
- objective
- projects
- public: true
- references
- resumes
- skills
- social



```
CreateResume.js
Pikachu.js x dayjs.js x index.js x gatsby.js x templateOptions.js x Templates.module.css x Templates.js x context.js x
1 import { MdAdd } from 'react-icons/md';
2 import { useTranslation } from 'react-i18next';
3 import React, { memo, useContext } from 'react';
4 import * as styles from './CreateResume.module.css';
5 import { handleKeyUp } from '../utils';
6 import ModalContext from '../contexts/ModalContext';
7
8 const createResumeButtonDataTestId = 'create-resume-button';
9
10 const CreateResume = () => {
11   const { t } = useTranslation();
12   const { emitter, events } = useContext(ModalContext);
13
14   const handleClick = () => emitter.emit(events.CREATE_RESUME_MODAL);
15
16   return (
17     <div className={styles.resume}>
18       <div className={styles.backdrop}>
19         <MdAdd size="48" />
20       </div>
21       <div
22         data-testid={createResumeButtonDataTestId}
23         tabIndex="0"
24         role="button"
25         className={styles.page}
26         onClick={handleClick}
27         onKeyUp={(e : KeyboardEvent<HTMLDivElement> ) => handleKeyUp(e, handleClick)}
28       >
29         <MdAdd size="48" />
30       </div>
31       <div className={styles.meta}>
32         <p>{t('dashboard.createResume')}</p>
33       </div>
34     </div>
35   );
36 }
```

# Code Snippets

## Code to create resume

```
ResumePreview.js
1 import { MdMoreHoriz, MdOpenInNew } from 'react-icons/md';
2 import { Menu, MenuItem } from '@material-ui/core';
3 import { navigate } from 'gatsby';
4 import { toast } from 'react-toastify';
5 import { useTranslation } from 'react-i18next';
6 import React, { useContext, useState } from 'react';
7 import dayjs from 'dayjs';
8 import DatabaseContext from '../../contexts/DatabaseContext';
9 import ModalContext from '../../contexts/ModalContext';
10 import * as styles from './ResumePreview.module.css';
11
12 const menuToggleDataTestIdPrefix = 'resume-preview-menu-toggle-';
13
14 const ResumePreview = ({ resume }) => {
15   const { t, i18n } = useTranslation();
16   const [anchorEl, setAnchorEl] = useState(null);
17   const { emitter, events } = useContext(ModalContext);
18   const { duplicateResume, deleteResume } = useContext(DatabaseContext);
19
20   const handleOpen = () => navigate(`/app/builder/${resume.id}`);
21
22   const handleMenuClick = (event) => {
23     setAnchorEl(event.currentTarget);
24   };
25
26   const handleDuplicate = () => {
27     duplicateResume(resume);
28     setAnchorEl(null);
29   };
30
31   const handleRename = () => {
32     emitter.emit(events.CREATE_RESUME_MODAL, resume);
33     setAnchorEl(null);
34   };

```

# Code Snippets

## Code for Resume Preview

```
Refactor Run Tools Git Window Help
Reactive-Resume-develop – database.rules.json
Add Configuration...
Git:
LeftNavbar.js LeftSidebar.js RightNavbar.js RightSidebar.js Hero.js DatabaseContext.js firebase.json
1 {
2   "rules": {
3     "resumes": {
4       ".indexOn": "user",
5       ".read": "auth !== null && query.orderByChild === 'user' && query.equalTo === auth.uid",
6       "$rid": {
7         ".read": "data.child('public').val() === true || data.child('user').val() === auth.uid",
8         ".write": "!data.exists() || data.child('user').val() === auth.uid || (!newData.exists() && data.chi
9       }
10    },
11    "users": {
12      ".indexOn": "isAnonymous",
13      "$uid": {
14        ".read": "$uid === auth.uid",
15        ".write": "$uid === auth.uid"
16      }
17    }
18  }
19 }
20 |
```

# Code Snippets

Code for firebase



# Issues:



Establishing proper  
connection with Firebase  
Realtime Database



Creating appropriate  
schema in the database



Issues with cascading style  
sheets while designing the  
website.

# Work Distribution

- Front end :

Anudeep & Jayanth

- Back end :

Jayanth, Manoj, Anudeep & Ravi

- Documentation :

Ravi & Manoj

# References

---

Wikipedia: <https://en.wikipedia.org/>

---

Smallpdf: <https://smallpdf.com/>

---

Resume Genius: <https://resumegenius.com/>

---

Firebase: <https://firebase.google.com/docs>

---

Article: <http://www.prosperoverseas.com/sample-resume>





Suggestions are appreciated....!!  
Thank you