Institution Management System

# System access

We can give access to

1. **Admin**: The soul authority to do things.
2. **Teachers**: The authority of educational data and access to all data.
3. **Parents**: They can access all the data of all their dependents.

**\* Providing authentication for every stakeholder needed a proper cloud service. Firebase could be better choice for us.**

***“I prefer to use email link authentication as it will make it more secure and easy for verification every time.”***

# System Levels

**Level-1:** Admin authentication. (Both **online** & **offline** methods)

**Level-2:**  Teacher authentication.

**Level-3:**  Parents authentication.

# Features

1. Activity logs.
2. Login & Registrations.
3. Notifications & reminders.
4. Student’s details & documents.
5. Staff details & documents.
6. Admission process.
7. Student’s payment data. (Fee categories, concession)
8. Institution’s transaction data. (vouchers)
9. Progress data of a student with link.
10. Teacher’s attendance.
11. Student’s attendance.
12. Tours & extracurricular activities.
13. Dashboard & Accountings.
14. Time table generation.
15. Syllabus planning.
16. Examination Planning. (Admit cards / Due list / Sitting plan)
17. Backup.

# Upgrade functions

# Security features

1. System Backup
2. Cloud Backup
3. Activity logs

# Activity logs

This records every action on the system. So that **no error occurs** & there will be a **transparency** in the system.

Are we going to have a single datasheet for all the activities?

Yes, we are doing. So, but with attaching the ID of the person from LOGIN sheet.

**Example**

Transaction’s sheet

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Amount | Mode | Sheet ID | Link ID | Activity ID (***FK)*** |
| 0001 | 5000 | SBI | STUD PAY | 005 | 0001 |
| 0002 | -2000 | CASH | VOCH PAY | 008 | 0002 |

Activity sheet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Comment | Prev ID | Time | User ID |
| 0001 | 5000 Paid by - to – for stud payment | - | 02:11:2021 ; 05:30:PM | 1 |
| 0002 | Given to – by – for voucher | - | 02:11:2021 ; 05:45:PM | 1 |

We have modified the first transaction to 10000 due to an error in entry.

**After Change**

Transaction’s sheet

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Amount | Mode | Sheet ID | Link ID | Activity ID  ***(FK)*** |
| 0001 | 10000 | SBI | STUD PAY | 005 | 0003 |
| 0002 | -2000 | CASH | VOCH PAY | 008 | 0002 |

Activity sheet

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Comment | Prev ID | Time | User ID |
| 0001 | 5000 Paid by - to – for stud payment | - | 02:11:2021 ; 05:30:PM | 1 |
| 0002 | Given to – by – for voucher | - | 02:11:2021 ; 05:45:PM | 1 |
| 0003 | 5000 modified to 10000 due to an error in entry | 0001 | 02:11:2021 ; 05:55:PM | 1 |

Here,

1. We have added a new activity log for the modification.
2. We have given the initial transaction’s activity log ID to new activity log as a pervious log ID.
3. We have updated the amount to 10,000 in both STUD Pay & transaction Sheet. We have updated the new log ID to the transaction.

**Logic:**

We can recursively get to know an activity logs of any transactions completely.

**Further expansion - Questions:**

How do we make the access levels for the user? Designing a separate design for them.

# Login & Registrations.

Here we have **three sheets based on user level**.

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Mail** | **Password** | **UserID** |
| 001 | [Rpranayreddy40@gmail.com](mailto:Rpranayreddy40@gmail.com) | 537537537 | 500 |
| 002 | [Rpranayreddy40@icloud.com](mailto:Rpranayreddy40@icloud.com) | 537537537 | 200 |

Here, **User ID is a foreign key to the stakeholder’s sheet.**

*Stakeholder’s sheet gives a unique ID to every user in the environment.*

**\*\*Password can be encrypted.**

# Notifications & reminders.

This sheet serves the purpose of both notification & reminder.

Where reminder is just a customized notification.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Created Time** | **Trigger Time** | **Iterations** | **Duration** | **Level** | **Action** | **Text** | **User ID** | **Done** |
| 08:05:00 AM; 02/11/2020 | 10:00:00 AM  01/01/2021 | -1 | 1 year | 0 | 005 | Happy New year. We wish…. | 100 | Yes |
|  |  |  |  |  |  |  |  |  |

Here,

**Created time:** time of creation.

**Trigger Time:**  When to pop the notification.

**Iterations:**  no of time it should be triggered. (“-1” is a loop iteration)

**Duration:**  duration of the iteration.

**Level:** This depicts its importance.

*0: only in the notification bar*

*1: pop up and stay for x mins and then disappear.*

*2: Action needed. (No close button available)*

**Action:**  link of the action (a new table can be created or cane be given a link here it self).

***But to have a simple action table. So that if needed to change a link address change be done in time.***

**Text:** Message.

**User Id:**  who should be getting the message.

***\*\*\*We can send a reminder to particular stakeholder at particular time. So that it works as a mail too. Then pop-up will not be action. But it will take us to notification. Then the entire message comes and the click will take to action***

***(or) Create a new similar table to reminders. \*\*\****

**Flow  
We need to create holidays table to have attendance in it. We have to not include holidays in the notification. Because it causes the problem when holiday is only for primary.   
So, we are just making holiday table also as notifications. Where notification system also checks holiday’s table.**

# Student’s details & documents

In order to join. We have to create them Tuples in these tables:

1. Stakeholder
2. Student details
3. Attendance
4. Student Balance
5. Progress

Basically, **Stakeholder table contains unique ID for each person.** *So, that when clubbing all person like in attendances sheet or notification sheet. Then there won’t be same ID for different users.*

Here, we create a **student details** table containing just photo.

**Student details** parameters:

1. ID
2. Full Time of Admission (to produce **dynamic links** for*progress of student*)
3. Stake ID
4. Name
5. Class
6. Roll no
7. Photo Id or so.
8. Fee structure excluding concession.
9. Merit-based concession
10. Parent ID’s (Two or three)
11. Other basic parameters

# Teaching & non-teaching staff details & documents

In order to join. We have to create them Tuples in these tables:

1. Stakeholder
2. Teacher details
3. Attendance
4. Staff Balance

Here, we create a **teacher’s details** table containing just photo.

**Staff details** parameters:

1. ID
2. Stake ID
3. Name
4. Date of birth
5. Phone number
6. Photo Id or so.
7. Salary
8. Other basic parameters

*If we need data of only teaching staff then we are going to make a new table.*

# Admission process

This will have two methods: **1. Admin**

**2. Website**

Both will **just do entry of parent and student’s details**.

*In both methods, Process is done only on completion of some parameters like Admission Fee, Exam & some of the other parameters.*

Initially, we will have only **Single Complete table** (*one table per category “Students, Parents, Staff”*). It contains all parameters needed of a new entry at their end.

***Note: Login & Stakeholder Table is same for both. Just data which should be cross-checked will be on hold in a different table.***

After Each step user has to **collect a copy of their details with office holder signature**. For further use.

# Student’s payment data. (Fee categories, concession)

This helps in generating a *quick sheet of fee due* & also helps in *validation of records*.

This is common for all students

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Stake ID** | **Fee excluding merit concession** | **Concession** | **Fee paid till now** | **Sports** | **Tour-1** |
| 1 | 100 | 15,000 | 1,000 | 10,000 | 500 |  |
|  |  |  |  |  |  |  |

**FLOW**

We have to make a Fee Structure table for class-wise. (That includes tours & etc.)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Title** | **Fee** | **Class-1** | **Class-2** | **Class-3** | **Class-4** | **Class-5** | **Class-6** | **Class-7** | ***Soon🡪*** |
| Sports | 500 | NO | NO | NO | Yes | Yes | Yes | Yes | Yes |
|  |  |  |  |  |  |  |  |  |  |

**(or)**

We can **go with Class ID or CODE Table** & then add fee structure for each.

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Title** | **Fee** | **Class Code** |
| 1 | Sports | 200 | P1 |
| 2 | Sports | 300 | P2 |
| 3 | Sports | 400 | P3 |
| 4 | Sports | 500 | P4 |
| 5 | Sports | 500 | P5 |
| 6 | Sports | 500 | P6 |
| 7 | Sports | 500 | P7 |

# Institution’s transaction data (vouchers)

This includes: 1. Teaching staff

2. non-teaching staff

3. Vouchers

4. Payment’s

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Date with time** | **Name or Stake ID\*\*** | **Phone number** | **Amount** | **Mode** | **Admin ID** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**\*\* Drop down should be appeared for the payment of any bills (petrol & etc.)**

**FLOW**

We have to **make a staff balance sheet also which I have not mention.**

# Progress data of a student with link

This is basically done by downloading computer based CSV and then upload it back.

This progress is based on category.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Stack Id | SA-1 (Telugu) | SA-1 (Hindi) | SA-1 (Maths) | SA-1 (science) |
| 1 | 100 | 95 | 95 | 95 | 95 |
|  |  |  |  |  |  |

**This progress reports can be downloaded for meet use (or) produced to parents on activation.**

**Doubt**

To *create a new table for F.A’s and S.A’s* **(or)** *just add new columns***.**