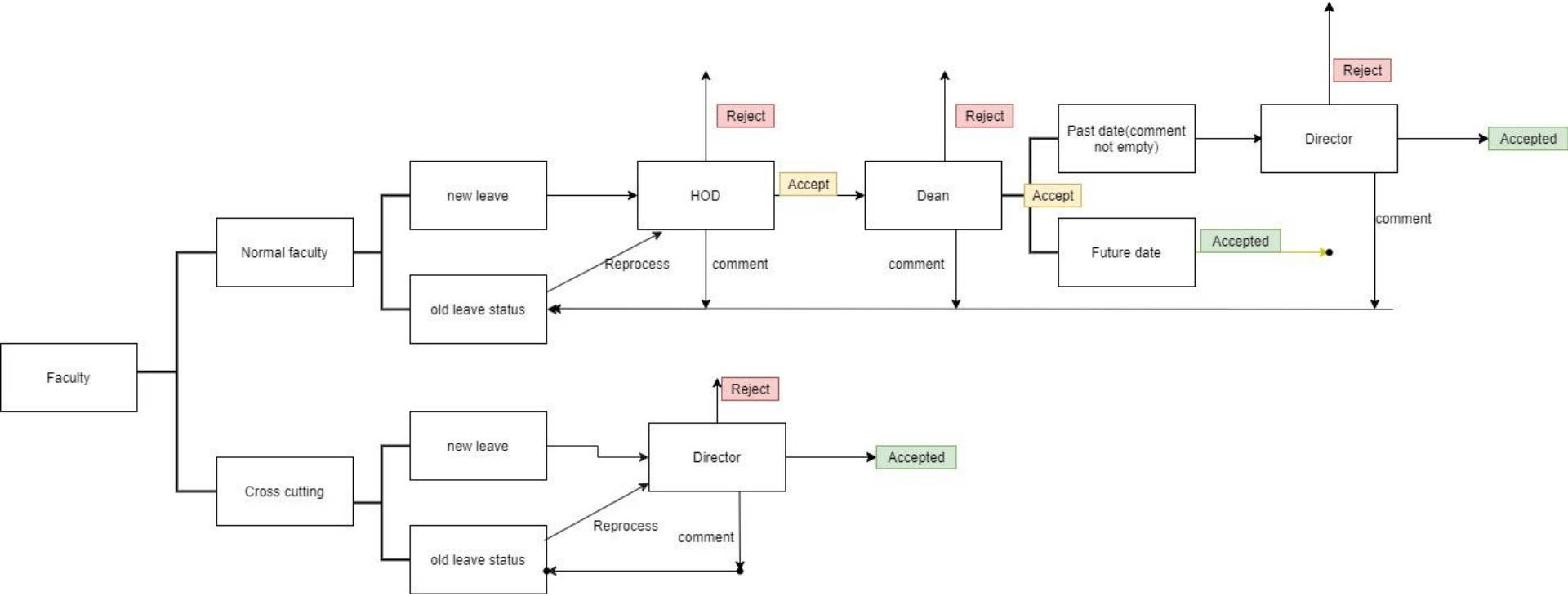
DBMS project UI and backend database

Ankur | Aman | Tarun

Link to the project folder - LINK

Flow Chart



Note: All colored boxes are also notified in ‘old leave status’.

Constraints:

1. Colored boxes are updated in old leave status.
2. In whole process, a trigger will be used - if at any step, the date for applied leave crosses the present date, the leave will be auto-rejected.
3. If a leave is in process, new leave cannot be processed.
4. If a faculty leaves institute, he/she cant apply for new leaves.
5. If faculty applies for more leaves than available, maximum possible leaves will be deducted.
6. Director can change Dean FAA and Dean SP, and all HODs.
7. If an HoD is changes, new HoD must be placed in his/her position.
8. Comments on an application are visible to every cross cutting faculty.
9. Projects can be created by any faculty, and details can be filled by PI or co-PI.
10. Approved projects are visible to every project head.

Will use 10 tables:

1. **Faculty (f)**
2. **Leave_in_process (lip)**
3. **Leaves_data (ld)**
4. **Leave_comments (lc)**
5. **Users**
6. **Current_users**
7. **Project**
8. **Project data**
9. **Budget**
10. **faculty_sp**

Table 1: Faculty

For data of whole faculties + hod + dean + director

Faculty_id	First name	Last name	email	department	DOB	gender	Position	pending_leaves	Status
							Faculty-3 HOD-2 Dean-1 Director-0	Initial leaves = 10	Active - 1 Inactive - 0

Table 2: leave_in_process

Gives data about leaves which are in process (does not contain outdated leave data)

Faculty_id	**Applied on**	**Leave_id**	**status**	**Directed to**	**Leave start date**	**number_of_leaves**
Connect table 1 and table 2	Date of application submission	Specific for each leave	Pending / approved / rejected / auto-rejected	HOD / Dean / Director		Leave for how many days?

Table 3: Leaves_data

This table has record of all (previous + current) leaves

Faculty_id	Leave_id	Status	Applied on	verdict_date	Leave start date	Number of leaves
		Accepted / rejected / pending / auto-rejected				

Table 4: Leave_comments

Includes data for comments for all leaves - previous + current

Leave_id	Comment_id	faculty_id	Comment by	Comment	Forwarded to	date_of_comment
			Faculty / hod / dean / director	A string	Hod / dean / director	Date of comment

Table 5: Users

For ease of setting up passwords, username is same as password in the default database we used, but it can be set anything by the faculty. A new faculty can register as well. Also admin can de-enroll a faculty as well.

username	password
rpr001	rpr001

Table 6: current_users

faculty_id
rpr001

Table 7: Project

Project_id	project_name	project_description	main_pi	co1	co2	co3	co4	co5

Table 8: Project_data

project_id	directed_to	status
	1 = main PI 2 = Dean SP	0 = Pending 1 = Approved 2 = Rejected

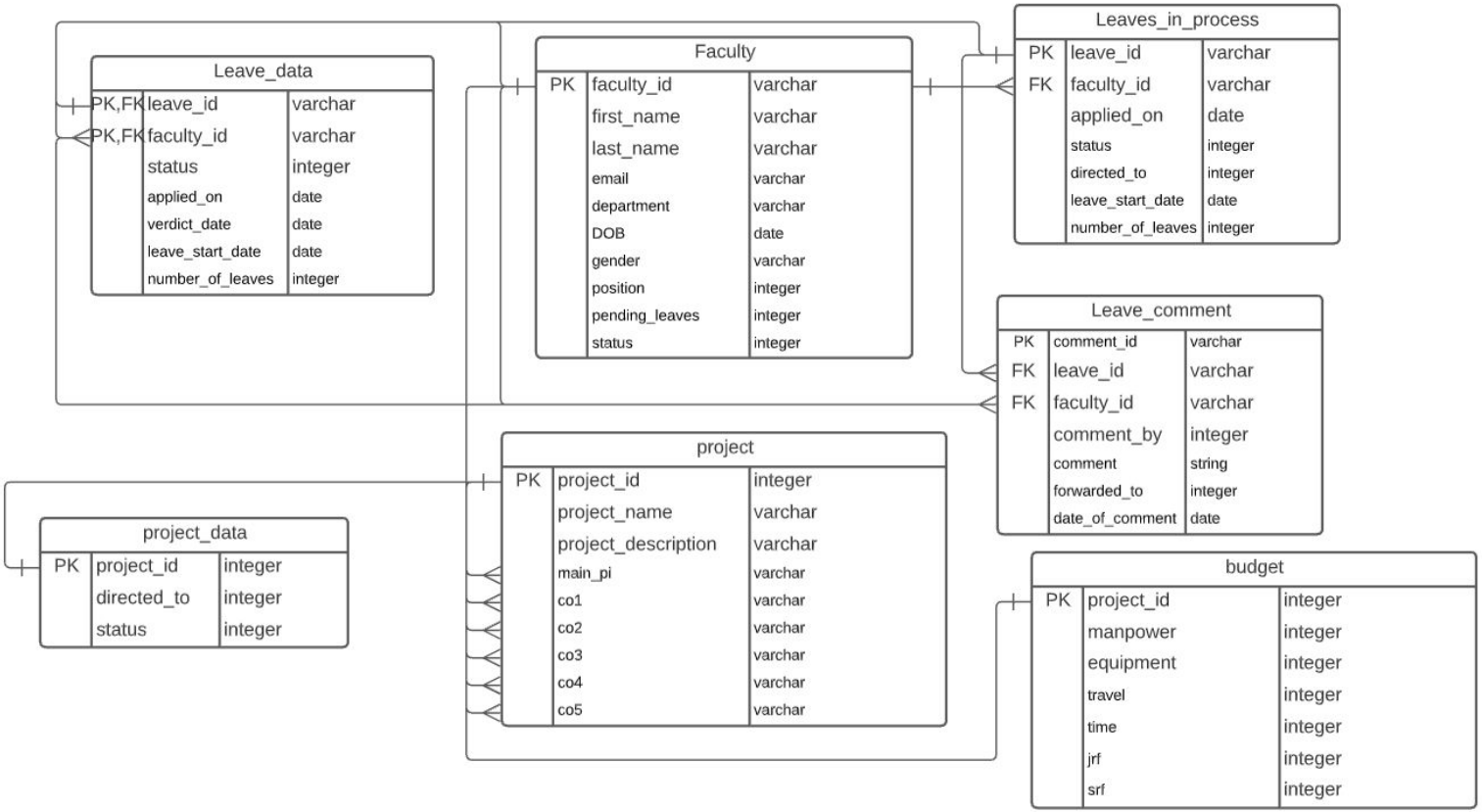
Table 9: Budget

project_id	manpower	equipment	travel	time	jrf	srf
	In Rs.	In Rs.	In Rs.	In months	Number of JRFs	Number of SRFs
					(Salary fixed)	(Salary fixed)

Table 10: faculty_sp

faculty_id
Faculty ID of only Dean SP

ERD


Leave_data
PK,FK leave_id varchar
PK,FK faculty_id varchar
status integer
applied_on date
verdict_date date
leave_start_date date
number_of_leaves integer
Faculty
PK faculty_id varchar
first_name varchar
last_name varchar
email varchar
department varchar
DOB date
gender varchar
position integer
pending_leaves integer
status integer
Leaves_in_process
PK leave_id varchar
FK faculty_id varchar
applied_on date
status integer
directed_to integer
leave_start_date date
number_of_leaves integer
Leave_comment
PK comment_id varchar
FK leave_id varchar
FK faculty_id varchar
comment_by integer
comment string
forwarded_to integer
date_of_comment date
project
PK project_id integer
project_name varchar
project_description varchar
main_pi varchar
co1 varchar
co2 varchar
co3 varchar
co4 varchar
co5 varchar
project_data
PK project_id integer
directed_to integer
status integer
budget
PK project_id integer
manpower integer
equipment integer
travel integer
time integer
jrf integer
srf integer

Use 5 interfaces UI

1. Login Portal
2. Faculty Profile Portal
3. Faculty Portal
4. HOD portal
5. Dean Portal
6. Director Portal
7. Projects Portal
8. Approve projects (only for Dean SP)

Faculty Portal

Login credentials - faculty_id + password

Profile	**Apply new leave**	**Old leave status**
Faculty id, Name, Dept, Age, Gender, Aadhar card, Univ status, No. of leaves pending	*Only active if old leave status NULL*	Old leave at which level? Comments made by HoD / Dean / Director
	Ask for - Date of start, number of days, comments	Provision to add comments and direct to anyone

HOD portal

Login credentials - faculty_id+ password

It will have a special section to approve, reject or send back leave applications with comments. Only those applications will be visible here which are supposed to be approved by HOD

Dean Portal

Login credentials - faculty_id + password

It will have a special section to approve, reject or send back leave applications with comments. Only those applications will be visible here which are supposed to be approved by Dean.

Dean sponsored projects and dean SA are different and have different portals.

Director portal

Login credentials - faculty_id + password

Director portal has a special previledge to change dean or HOD at anytime. Also leave application approval portal will be visible.

UI

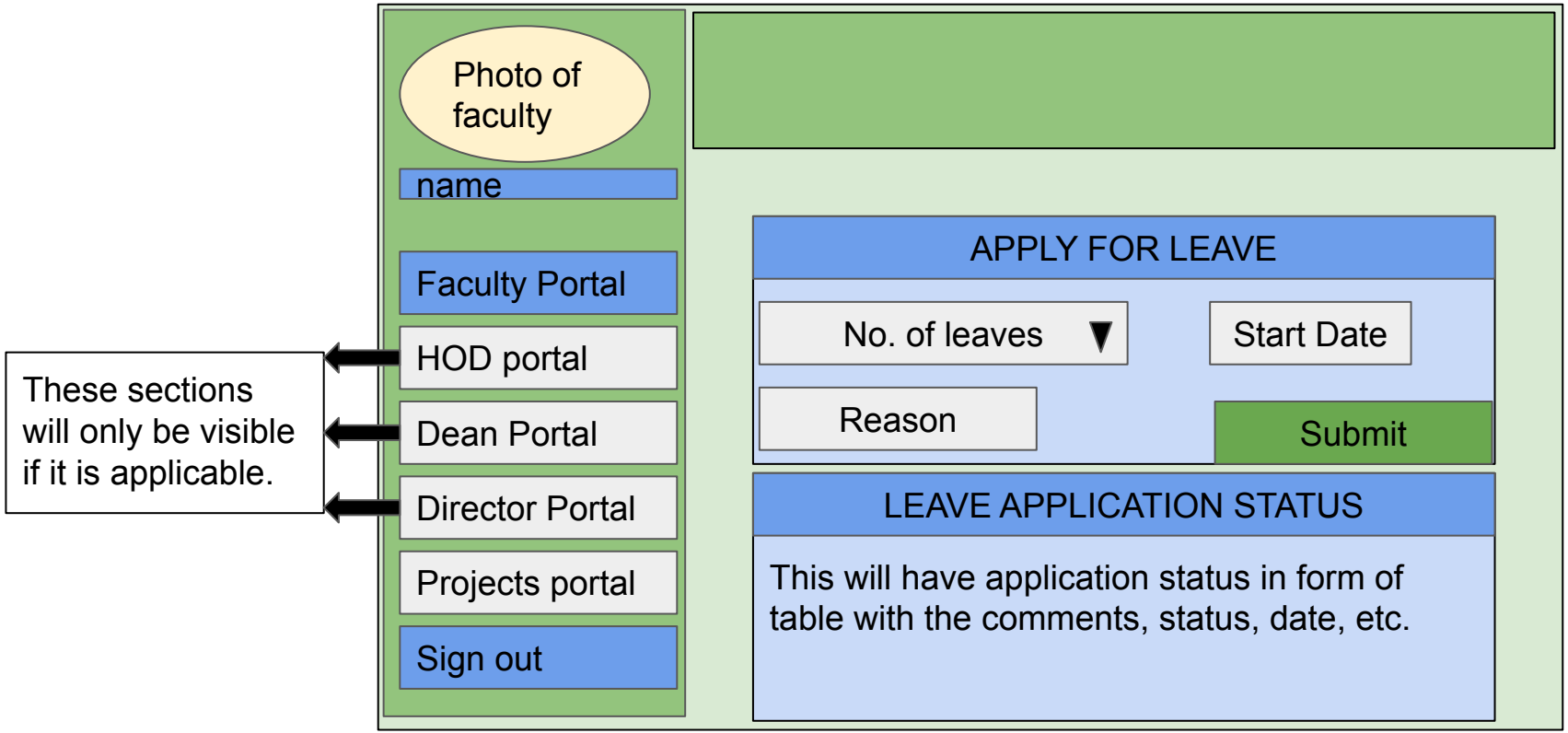


Photo of faculty
name
Faculty Portal
HOD portal
Dean Portal
Director Portal
Projects portal
Sign out
These sections will only be visible if it is applicable.
APPLY FOR LEAVE
No. of leaves
Start Date
Reason
Submit
LEAVE APPLICATION STATUS
This will have application status in form of table with the comments, status, date, etc.

	Faculty		HoD			Dean			Director		
	Apply	Status	Apply	Status	Approve/ reject / back	Apply	Status	Approve/ reject / back	Apply	Status	Approve/ reject / back
Retrieve data from	Faculty	Lip Lc Ld	Faculty	Lip Lc Ld	Lip Lc	Faculty	Lip Lc Ld	Lip Lc	Faculty	Lip Lc Ld	Lip Lc
Clicking on button will trigger changes in	Lip Lc Faculty	Lip Lc Ld final verdict	Lip Lc Faculty	Lip Lc Ld final verdict	Lip Lc Ld, if reject	Lip Lc Faculty	Lip Lc Ld final verdict	Lip Lc Ld	Ld	-	Lip Lc Ld

Lip = Leave in process, Lc = Leave_comments, Ld = Leave Data

Cases

1. Faculty / HoD / Dean / Director applies for leave
2. Faculty / HoD / Dean comments in status portal
3. HoD approves / rejects / revert backs the leave request in approve portal
4. Dean approves / rejects / revert backs the leave request in approve portal
5. Director approves / rejects / revert backs the leave request in approve portal
6. Auto-reject for retro leave applications
7. Faculty applies for a project
8. Main PI or co PI could fill the details of project
9. Dean / Main-PI can approve or reject project

Case 1: Click is on ‘apply’ button

If faculty.position < 3: Add an entry in Lip, Lc (if any comments attached) and change number of pending leaves in faculty.

Else: add an entry in Ld, Lc (if comments attached) and change number of pending leaves in faculty.

Case 2: Change in ‘status’ portal - comment button clicked

add an entry in Lc, update entry in Lip - in all cases, irrespective of faculty.position

Take care of directed_to variable.

Imp: This portal won’t exist for director.

Case 3: Approve / reject / revert back by HoD

If hod approves: Edit entry in Lip, Add entry in Lc (if comments attached).

Else if hod reverts back: update Lip, add entry in Lc.

Else if hod rejects: create an entry in Ld, delete entry from Lip, add entry in Lc (if comment attached).

Case 4: Approve / reject / revert back by Dean

If dean approves: 2 cases
If future date leave: Add entry in Ld, delete entry in Lip, Add entry in Lc (if comment)
If past date leave: Update Lip, Add entry in Lc (if comment)

Else if dean reverts back: update Lip, add entry in Lc.

Else if dean rejects: create an entry in Ld, delete entry from Lip, add entry in Lc (if comment attached).

Case 5: Approve / reject / revert back by Director

If director approves: Create entry in Ld, delete entry from Lip, Add entry in Lc (if comments attached).

Else if director reverts back: update Lip, add entry in Lc.

Else if director rejects: create an entry in Ld, delete entry from Lip, add entry in Lc (if comment attached).

Case 6: user signs up

New user signs up, store data in users table

Case 7: User logs in

Check values from users table

Clear current_users table

Edit / Insert new entry in current_users

All time triggers

Trigger1: Auto-reject if future leave and today date = leave date: create and entry in Ld, delete entry from Lip

Trigger2: Keep scanning Ld, for final verdicts of all leaves app

Steps to link php and database

1. Download XAMPP
2. Open xampp-control and connect apache and mysql
3. In local browser, open localhost/phpmyadmin
4. Create new database - dbms_project
5. Use code generated by lucid charts to generate tables
6. Create csv file of faculty sheet and import data in faculty table

continued

1. In xampp, there must be a folder htdocs.
2. Delete existing files in htdocs
3. Create an index.php file and a new folder ‘includes’
4. Inside ‘includes’ folder, create an dbh.inc.php (database handler) and ak.sql (command calling) files
5. Code for all three files in upcoming slides

IMP: Design algo in terms of stored procedures and call those functions in .sql files, when needed.

Stored procedures

There are 10 stored procedures used in this project.

They can be found in the folder shared with codes of the whole project with names stored_procedures_x.sql

Stored Procedure name	Input variables	Functions
submitbutton	Faculty_id, leave_start_date, no. of leaves, comment	Create a new leave
nextbutton	Leave_id, comment, action_taken (0-reject, 1-forward/accept, 3-revert back)	Approve / reject / revert back application
checkyearend	-	Update leaves at end of year
addcomment	Leave_id, comment	Add a new comment to application
autodelete	-	Add back leaves for a faculty, if leave is rejected
change_por	Faculty_id, identifier	Change dean or HoD by director
createproject	Name of project, description, main, co1, co2, co3, co4, co5	Create a new project
apply	Faculty_id (logined), project_id, manpower, equipment, travel, month, srf, jrf	Fill the details of project
approve_pi	Project_id, decision (1-approve, 2-reject)	approve / reject by main PI
approve_dean	Project_id, decision	approve / reject by dean SP