# ADD/DROP ONLINE FORM PORTAL

# (DESIGN DOCUMENT)

## What is it?

In this project, we have made a portal for adding and/or dropping courses online.

All the required processes viz. filling of the form by the students, approval by the teacher(s) viz. course instructor(s), course advisor(s) would be done online.

Students are now just required to log on to the portal and fill the form which would be further given approval by the required teachers.

## Problem Description and Motivation for the project

- The students sometimes feel lethargic toward the work of adding and/or dropping the courses of the curriculum.
- Moreover the problems like unavailability of the required teachers viz. course instructor(s), course advisor(s) or sometimes the academic section staff, makes the work hectic.
- Also most students rush on the last date to do the work which overloads the work for all on that particular day.

## Design Issues

- Logging in via Gmail or creating own database?
- Database design:

  - a single table for students of all the year.

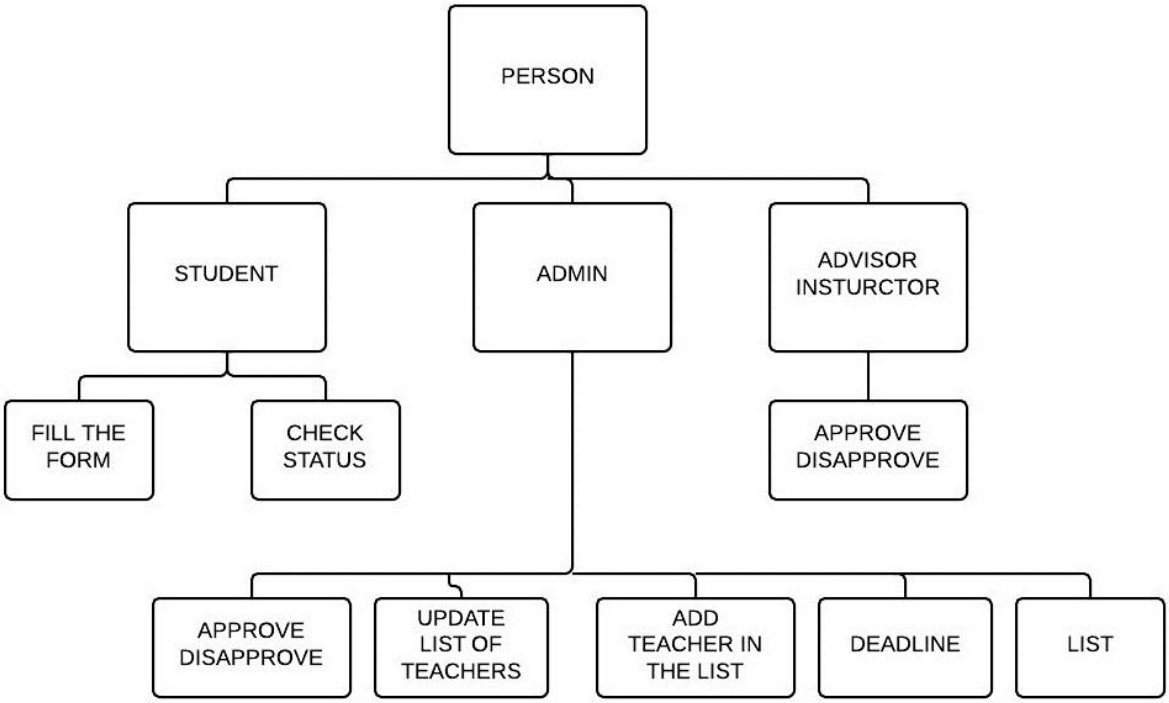
  - different tables for students of every year.
- Should there be a mail sent to the teacher or should there be a particular portal for the required process of adding/dropping?
- Restrict logging in to the portal to only iitrpr?
- Changing of instructors for subjects.

- Security.
- Approval from academic section.

## Design Decisions

After figuring out final ways to completely accomplish the designing of the project, following things have been done:

- A portal for students, teachers to log into and do the required work.
- Separate pages for different users.
- Once the students have filled the add and/or drop forms, the input data will be updated in the database.
- Once the deadline is over, the course instructor(s) and course advisor(s) could perform their action.
- Authentication using Google api oauth 2.0. because then there won't be any need for additional user ids and passwords.
- Only iitrpr users can log in.
- Once a person logs in, we identify whether it's a student or a teacher.
- If a student logs in, his form details will be written to the database using PHP.
- If a teacher logs in, all the course add/drop requests which were under his/her name, will be shown to him for his/her approval/rejection.
- Since security was a big concern, the corresponding pages will be not be rendered unless the correct person logs in.
- Sessions are created by the API to avoid re-logging.
- If academic section staff logs in, they are shown a portal where they can approve or reject courses and change instructors for subjects.
- The corresponding data from the PHP pages is then updated into the databases accordingly as required.



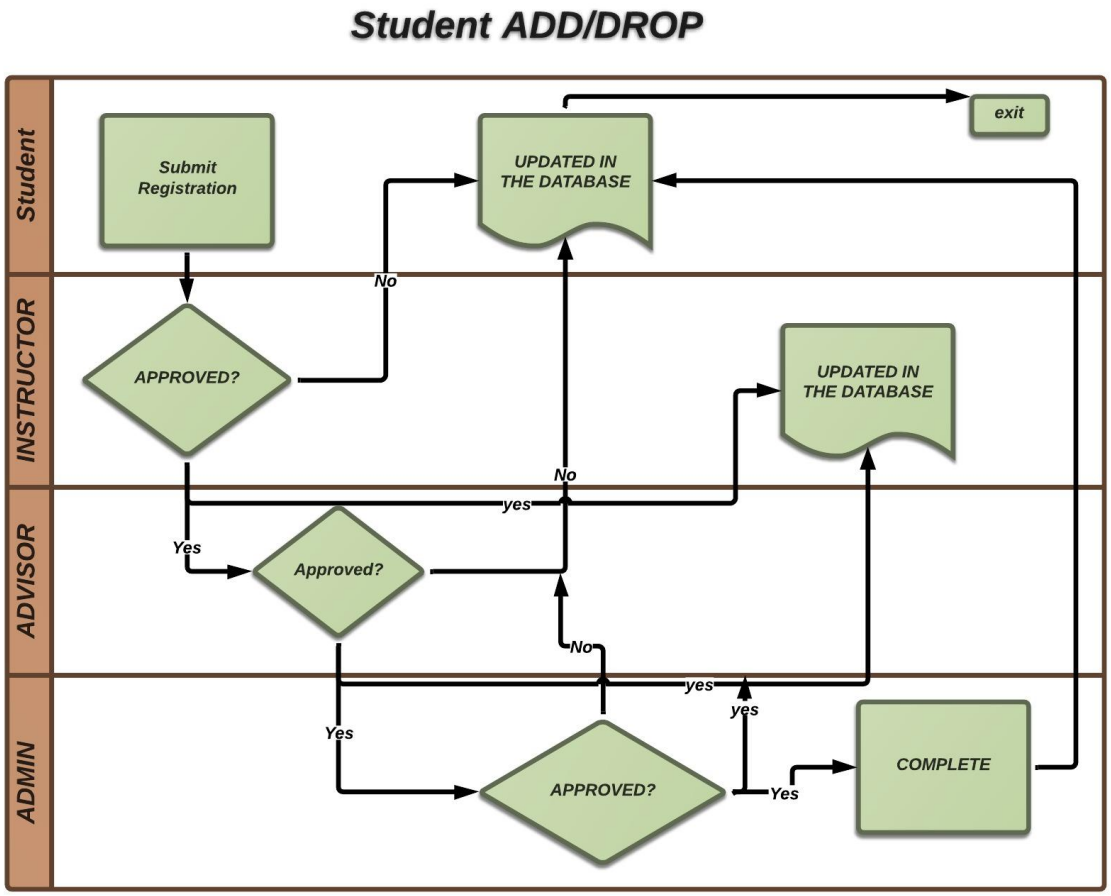

DIAGARM-:Different actions a user can perform

The above diagram explains the different actions a user can perform.

## Why was this design chosen?

- We didn’t want users to make one more user account and password so instead we made the login system using google API.
- Numbers of students in our college is not very high,certainly nothing that the database can’t handle.Hence instead of making table for each year we just created one table for all the students.
- We didn’t want students or instructor’s mail to be flooded so we used a portal where students and instructors can login and perform their actions.
- We didn’t want anyone of outside iitrpr to login.Hence we restricted the domain to iitrpr.
- We didn’t want students to login and see the advisor/instructors’s page.Appropriate security checks have been implemented to prevent the same.

# FLOWCHART/WORKING OF PROJECT




- We see a portal page that can be used to login by the users which include the admin, teachers, course advisors and the students. They will be able to login only if their email id belongs to iitrpr domain. Now, their individual information is saved in a database and each time a user logs in, the corresponding page opens up.
- A student will be able to apply for adding/dropping a course or he/she can check the status of any course that he has applied for previously.
- The corresponding course instructor can approve/reject the students' request.

- If approved by the instructor, the course advisor can then approve or reject the course.
- If approved by the advisor, the admin/academic section will be able to approve/disapprove the course after evaluating all the criteria.
- The academic staff can also update teacher's information in the database, update the deadlines for the form submissions and change the teacher for a particular subject.
- If at any point the request is rejected, it does not go further for approval/rejection.

## Implementation Tools:

- Operating System: Windows 7.
- Text Editors: Sublime Text, Notepad++, Floobit for collaborative editing.
- phpMyAdmin for databases management.
- jQuery.
- Oauth 2.0.
- Composer for Oauth 2.0.
- json for data retrieval.