



# CODEDUEL

## CLIENT-SERVER MODEL

**N. Ajay Raj**

**M. Raghu**

**N. Amrutha Lakshmi**

**M. Sushma**

## PROBLEM STATEMENT

- ▶ The objective of project is to build an application for the event "Code Avadhan".
- ▶ In this event, the contestants are put in a duel and can be won by solving problems imposed by other contestant.
- ▶ The scoreboard should be limited to two contestants per view only.
- ▶ The problem should only appear when the opponent challenges us or vice versa.

## EXISTING SYSTEM

- ▶ The existing system(s) such as HackerRank, HackerEarth etc are suitable for hosting coding competitions.
- ▶ All the questions are visible to all the users which is not desirable.
- ▶ The entire leaderboard is visible which is not correct.
- ▶ Limited to stdin, stdout and stderr.
- ▶ Event will be conducted by manually sharing the files.

## PROPOSED SYSTEM

- ▶ The proposed system is designed to match the requirements of the event.
- ▶ Limited Leaderboard.
- ▶ Question will be shown after accepting challenge only.
- ▶ Access to file-system and database on the Server-Side (Limited Access to protect the server).

## FUNCTIONALITIES

- ▶ Limited Scoreboard
- ▶ Whenever the contestant requests to view the scoreboard, his contestant id will be sent to the server.
- ▶ On the server, his contestant id is matched against the database to find the opponent's id.
- ▶ The score is computed by adding an aggregate function on the database for these two ids and is sent back to the client.

## FUNCTIONALITIES

- ▶ Question available after accepting challenge only.
- ▶ Since this is a duel event, one contestant will challenge the other contestant. He will speak out a challenge key to his opponent.
- ▶ The opponent will enter the challenge key and accept it.
- ▶ Only after accepting, he will be able to view the question.
- ▶ This way the the opponent will not know the forth coming challenges posed by the contestant.

## FUNCTIONALITIES

- ▶ Test Run
- ▶ Once the contestant solves the problem, he will push the code to the server.
- ▶ The server will auto-detect the language based on the extension.
- ▶ A subprocess is invoked to run the submitted code.
- ▶ The test cases corresponding to the problem are run against it to find the correctness of the program.

## FUNCTIONALITIES

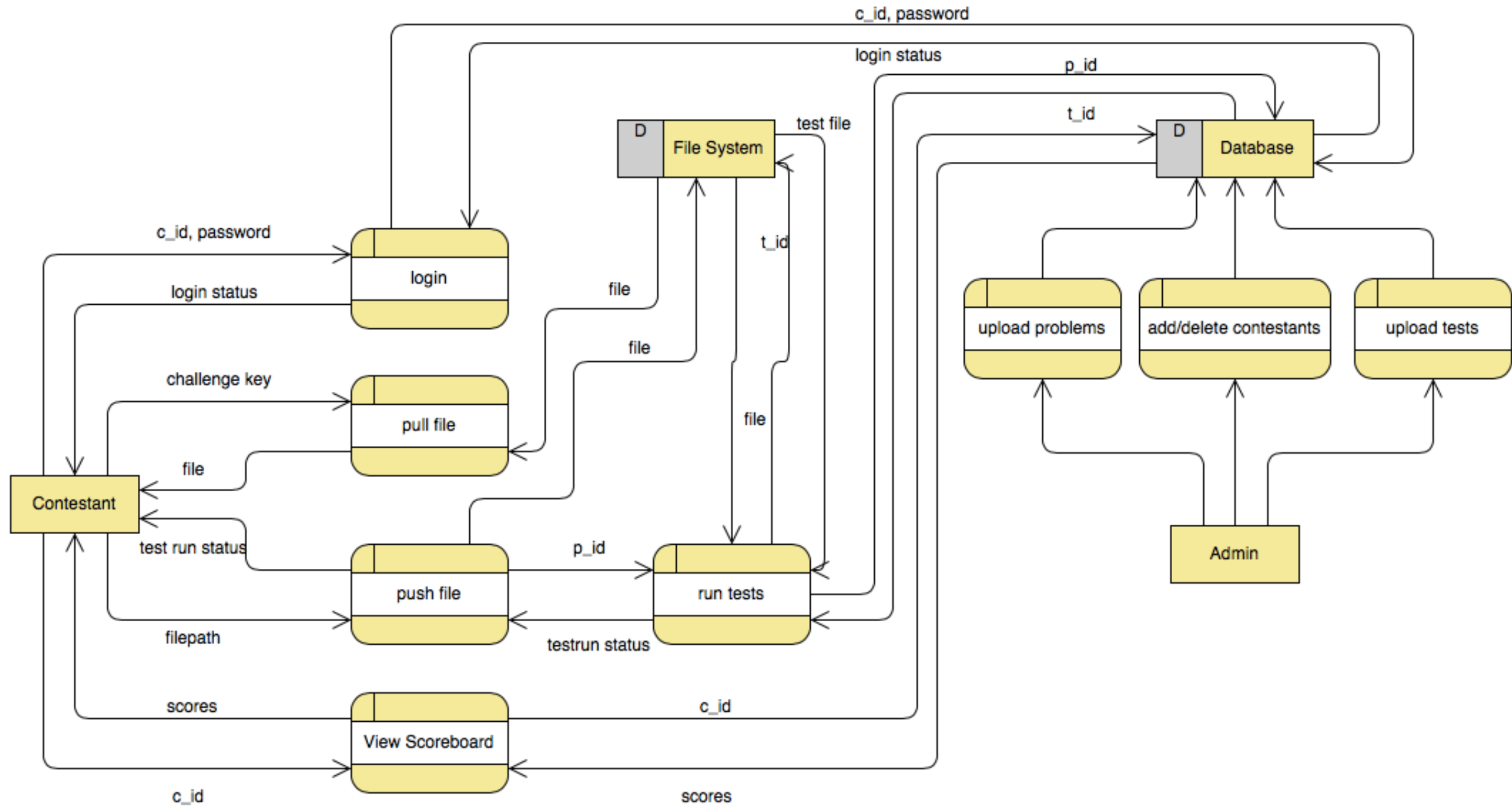
- ▶ Consists of a file system and a database.
- ▶ The file system will have 3 directories namely spec, src and test.
- ▶ The spec will contain the problem statements.
- ▶ The src will contain all the submitted problems.
- ▶ The test will contain files that are used as test cases.



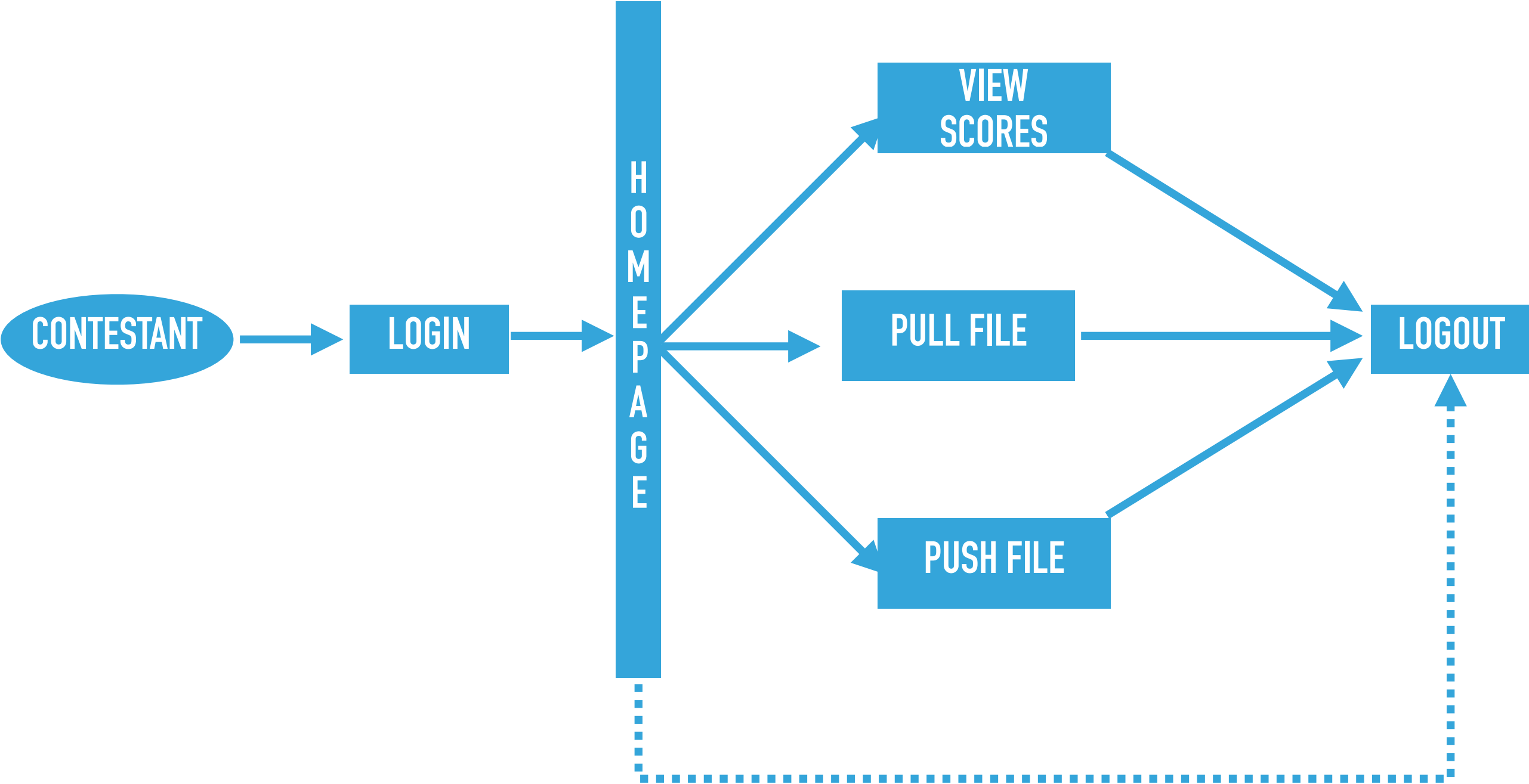
## FUNCTIONALITIES

- ▶ The Administrator Account.
- ▶ The Admin will have a separate web app to login.
- ▶ This account has rights to add / delete problems, contestants and test cases as well.
- ▶ This is the only account where the entire leaderboard is visible.

## DATA-FLOW DIAGRAM



# ARCHITECTURE



**THANK YOU**

**Ajay, Raghu, Amrutha, Sushma**