This is the use case you are going to implement in this exercise: “As an employee, I must be able to apply for leave against a specific leave type.”

The current implementation of the “leave management system” supports only one leave type: “Earned Leave”.

In this exercise, you are going to do an enhancement to the system: We are going to support one more leave type is called “Optional Leave”

 Optional Holidays are:  Employees should choose only 2 optional holidays

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Occasion** | **Date** | **Day** |
| 1 | Id-Ul\_fitr | 15,Jun,18 | Friday |
| 2 | Ganesh Chaturthi | 13,Sep,18 | Thursday |
| 3 | Dussera | 19,Oct,18 | Friday |
| 4 | Christmas | 25,Dec,18 | Tuesday |

When applying for leave, the employee needs to specify the leave type as well. Separate leave balances are maintained for each leave type and employee optional leave should be checked to ensure that only two optional leaves are applied in a year. The manager also needs to see the leave type while approving/denying the application.

The database design needs to change, the mappers and java logic, and rest interfaces and the ui code – the detail page, the leave application page, the leave details list and the pending leave applications list.

Points to note:

* Keep the code changes local to your desktop; do not push the changes to remote
* You need not fix the cli for this change; but extra points if you are able to fix the cli as well
* Extra points for additional tests related to this functional enhancement
* Do it using all the agile concepts you have learned:
  + Enter the user story in TRELLO.
  + Create sub-tasks for each work item that needs to be done
  + Progress the task through the swim lanes as you complete them