INTERNSHIP PROJECT

SELF HEALING OF A DATABASE

DONE BY,

SASHANTH.V

INTRODUCTION TOWARDS SELF HEALING

- Self-Healing is referred to as the ability of automated tests to automatically adapt changes in the database schema/ the data without having manual intervention.
- Improve efficiency by accelerating the error detection by automatically detecting and correcting errors.
- Use AI and ML techniques for identifying and correcting issues arising at the database.

WHY NEEDED IN DATABASE TESTING

Self-Healing is needed in database testing as there can be many problems arising due to:

1) Renaming of Columns
2) Change in the table structure
3) Data formats change

So, in this process they SQL queries get auto adjusted with accordance to the database columns and they attempt to fix the test case/ queries based on the past patterns.

BENEFITS OF SELF HEALING

Applying self-healing in a database provides us a lot of benefits l

1) Reduces the need of manual testing as it is usually time consuming when the number of rows present is high.
2) Plays a major role in improving test resilience.
3) Helps in reducing the errors due to change in schema or data change.

TOOLS USED FOR MY PROJECT

1) Python+unittest- Automating the test cases
2) MySQL connector- Connection of database
3) LLAMA via Ollama- AI model that fixes broken SQL queries on the fly

PROCEDURES INVOLVED IN MY PROJECT

1. DESIGN OF A DATABASE:

For designing a database I decided to go on with the creation of a table nam
players in a database named original_ipl. It has columns like:

1. id(Primary Key for its uniqueness)
2. name
3. age_at_selection
4. salary_in_cr
5. nationality
6. ipl_team
7. matches_played

This database is used only for viewing and another DB named csk_pla
created for testing purpose.

SCREENSHOT OF OUR DATABASE

Select MySQL 8.0 Command Line Client

```
mysql> select * from players;
+----+------------------+------------------+-------------+-------------+---------------------+----------------+
| id | name             | age_at_selection | salary_in_cr | nationality | ipl_team            | matches_played |
+----+------------------+------------------+-------------+-------------+---------------------+----------------+
|  1 | Shane Watson     |               34 |         9.5 | Australian  | Chennai Super Kings |             43 |
|  2 | Dirk Nannes      |               30 |         2.5 | Australian  | Chennai Super Kings |              5 |
|  3 | Michael Hussey   |               36 |         2.5 | Australian  | Chennai Super Kings |             64 |
|  4 | Matthew Hayden   |               37 |           4 | Australian  | Chennai Super Kings |             32 |
|  5 | Doug Bollinger   |               29 |           3 | Australian  | Chennai Super Kings |             41 |
|  6 | Ben Hilfenhaus   |               28 |           2 | Australian  | Chennai Super Kings |             21 |
|  7 | George Bailey    |               30 |         1.5 | Australian  | Chennai Super Kings |              4 |
|  8 | Josh Hazlewood   |               30 |           2 | Australian  | Chennai Super Kings |             12 |
|  9 | John Hastings    |               27 |           1 | Australian  | Chennai Super Kings |              2 |
| 10 | Jason Behrendorff |               31 |         1.5 | Australian  | Chennai Super Kings |              5 |
| 11 | Nathan Ellis     |               30 |           2 | Australian  | Chennai Super Kings |              1 |
+----+------------------+------------------+-------------+-------------+---------------------+----------------+
11 rows in set (0.00 sec)

mysql>
```

TESTINGS INVOLVED IN OUR PROJECT

For validating the self-healing in our project, we created a few testcases and they include:

1. Salary- The Salary must be non-negative
2. Nationality- In our database all the individuals must have the nationality as Australia.
3. IPLTEAM- In our database all the individuals must have their iplteam as Chennai Super Kings.
4. Duplications- Our database must not have any duplicate valu inserted and if present the value with the lowest id must be retained and the other value must be deleted.

TESTINGS INVOLVED IN OUR PROJECT

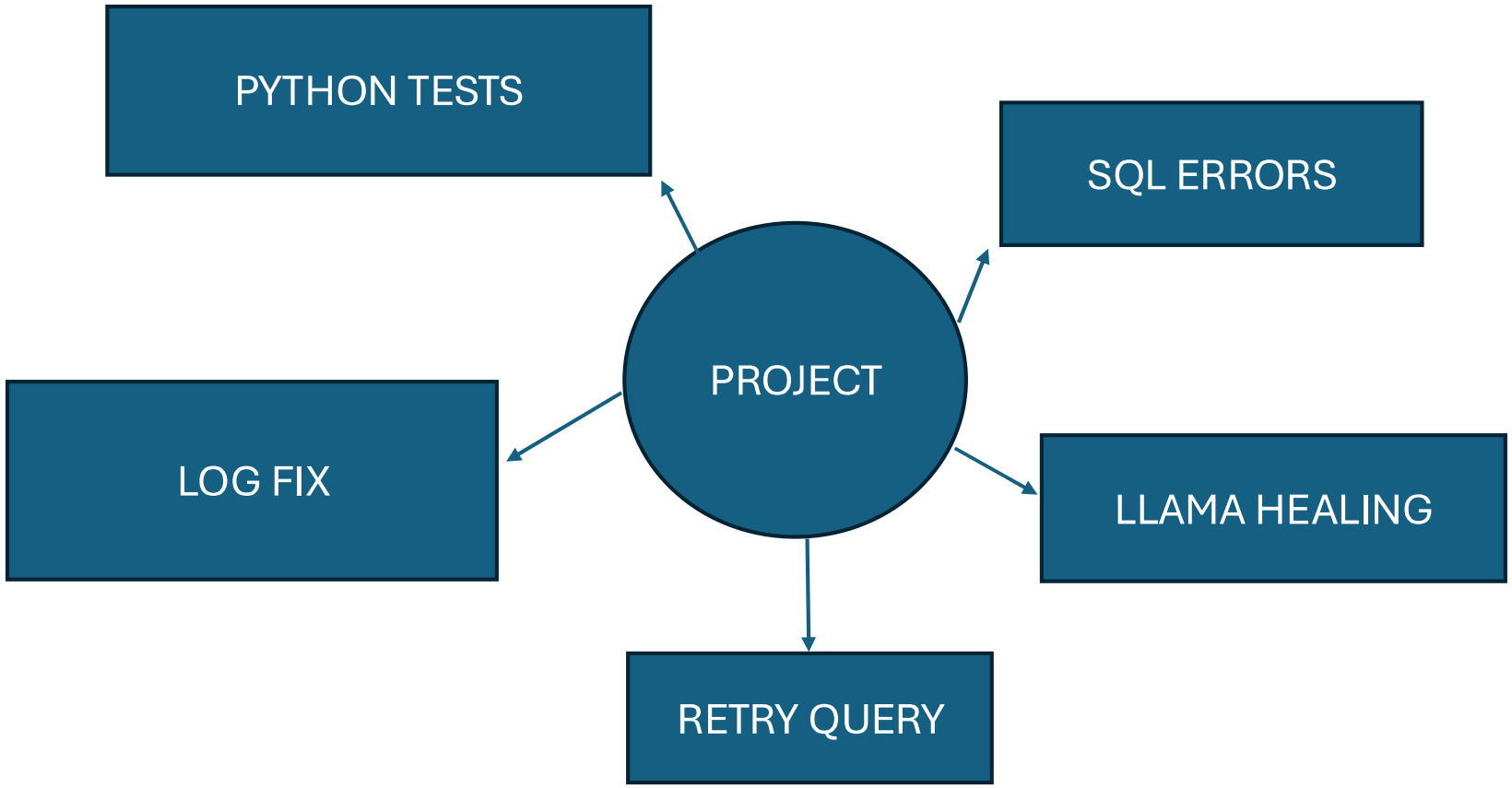
5. Testing must work even when there is a change in the name of salary column.

6. Testing must work even when there is a change in the name of table.

7. Usage of Memcached for storing the datatype of each column our DB AND compare it with the testing DB to check whether any change in datatype is present. If yes, identify it and then return it back to the original datatype.

8. Test the total number of columns in the testing DB. If there is a extra columns it must ask user permission to whether we must remove it or not. If yes, it will remove it else it retains it.

FLOWCHART OF OUR PROJECT



FURTHER IMPROVEMENTS

- In the future we could try to integrate into CI/CD(Combined pra continuous integration and continuous delivery) pipeline whic reduces manual intervention, and it is collectively called as co development or continuous software development.

- We could also use python coverage which shows us the percenta rows that were validated and percentage of issues that are healed.

- We could also try to have Role Based Access/ Permiss dashboard where admins would approve the healing and the teste only view.

RESULT

THUS, THE SELF HEALING PROCESS IN A DATABASE HAS BEE
OBSERVED SUCCESSFULLY AND THIS INDICATES THAT OU
FRAMEWORK CAN AUTOMATICALLY DETECT AND HEAL DA
ISSUES USING AI.

THE END