CHAPTER 5

IMPLEMENTATION

5.1 LIST OF MODULES

The project is implemented in Python programming language. The project is divided into 3 modules-

1. The functions module- This module implements the core and the additional functionality of the file.
2. The web application module implements the logic for handling user requests received over the web.
3. The UI module contains the front-end UI design and styling.

5.2 MODULE DESCRIPTION

1. functions module:

The functions model implements the main logic and handling operations needed to perform file structure operations. This model consists of functions.py file that contains all the logic functions like insert, update, delete and search functions including the hash function.

2. Web application module:

The web application module(app.py) implements the connection to functions.py, processes user requests received over the web and replies to them. This module is implemented in Flask, a micro-web framework for Python. The units of this module are represented as routes and their respective views, where the route describes the format of user request and the views implement the corresponding processing operations and response mechanism.

3. The UI module:

The UI module is divided into templated and styles. The templates contain HTML web page to be sent in response to the respective user requests. Styles contain the styling parameters for the web pages designed in templates.

Algorithm:

1. Insert: The insert algorithm accepts the three data that has to be inserted in the file along with the file name in which the file has to be inserted. The primary key (game name) is sent to the hash function and the position is calculated and seeked in the respective file. The other details, along with the primary key are added to the file at obtained position.

- 1. Start.
- 2. Input the details.
- 3. Compute the position by sending game name to the hash function.
- 4. Open the respective file and seek the obtained position.
- 5. Check if the position in the file is free.
- 6. If the position is free insert the new record in the file after padding.
- 7. If the position is occupied find if the record is a duplicate.
- 8. If the record is duplicate send appropriate message.
- 9. Else if the record is not duplicated and there already exists a record.
- 10.Then go to next hash location and repeat the above steps.
- 11.End

2. Search: The search function accepts the game name as primary key and then searches the respective file for the required data.

- 1.Start.
- 2.Input the Game name (Primary Index)
- 3.Send the primary key to the hash function.
- 4.Open the respective file and seek the location.
- 5.If a record exists in the obtained location check if the record has same primary key
- 6.If the record is found display the same if not display appropriate message

3. Delete: The delete function accepts the Game name user is looking for. It then searches the respected file for the Game name specified by the user. If present, then the record details are deleted from the file. Subsequently, the space is reclaimed in the memory. If it is not, then appropriate message is displayed for the absence of the record.

- Start.
- Input: Game name (Primary Index).
- Open respective file in read mode.
- Search for the record.
- If the record exists the record is deleted and the space is reclaimed.
- If the record is not found then appropriate message is displayed.
- End.

4. Display: Display function fetches the data from the respective file and displays it on the user interface.

- Start
- Read the required file from the respective file.
- Display the same.

5. Update: The modify function accepts the Game name the user is looking for. It then searches the file for the game name specified by the user. If present, then the record details are deleted from the file. Subsequently, the space is filled with the new record the user enters. If it is not, then appropriate message is displayed for the absence of the record.

- Start.
- Input: Game name(Primary Index).
- Search the file for the record.
- If found delete the record.
- Insert the new modified record in the same location
- Display the new record.
- End

6.Hashing: This function takes the primary key the game name and produces a hash value.

- Start
- Accept game name as input.
- Calculate Hash value.
- Return Hash value.