

# DSA

project  
"MITRATA"



- ▼ Khagendra Karki
- Divya Shangkat Karki
- ◀ Kripesh Nihure
- ▲ Avahan Tamrakar

# OBJECTIVES

- Implement data structures and algorithms into real life scenarios
- To learn and implement database for an application
- Experiment & learn about data structures
- To learn to use front end framework like react
- Enhance teamwork skills, utilize git
- To learn to use front end framework like react



# INTRO

## What is mitrata?

Mitrata is be a web-based app which provides the user with a platform to get to know other people and communicate with them. It is a social networking site like any other sites that exist such as bumble or tinder.

Most of us must have heard about sites/apps like tinder, bumble, grindr etc. Mitrata falls into the same category

The main motive of this app is to help people find a friend and explore if they do have some compatibility or not.



# WORKFLOW

In short, the workflow can be explained as follows:

Mitrata is a web application, which is a social networking platform.

Here a user is created from signup and logged in from log in.

The user data is stored inside the python based backend which utilizes the data structure log structure and merge tree.

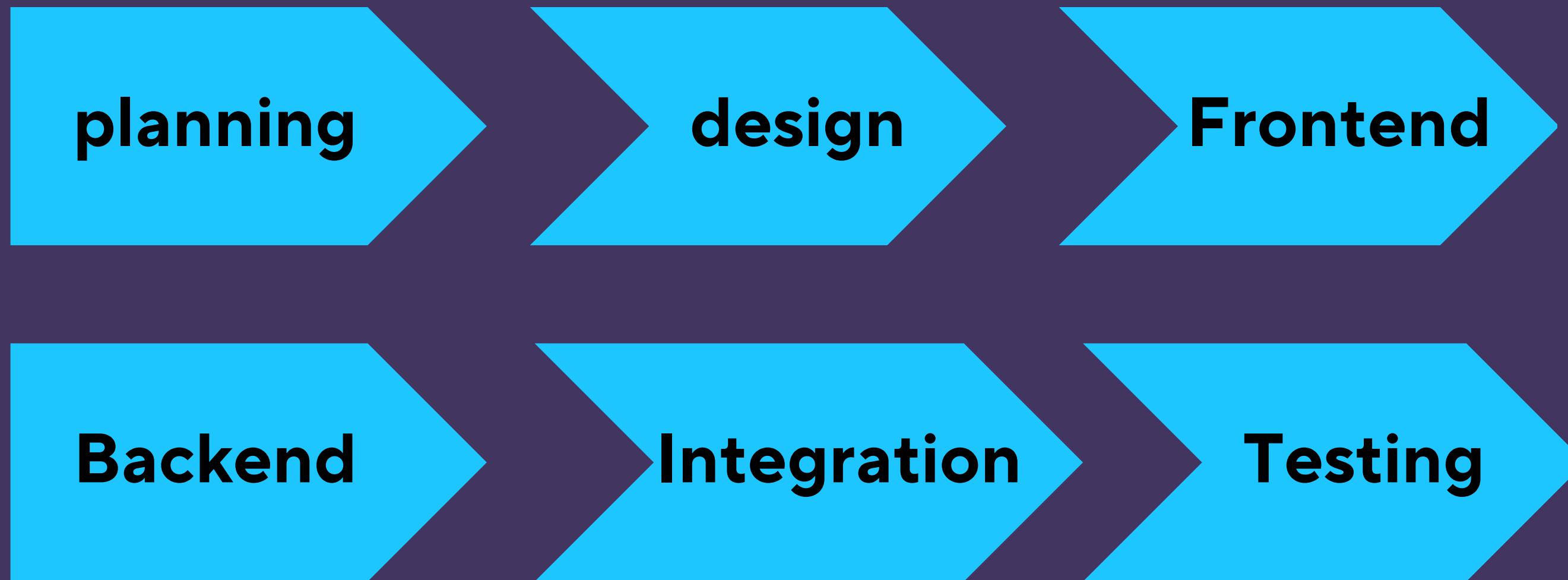
After getting logged in, user is able to swipe profiles right or left to send a friend request or move on to the next profile.

The request is sent to the notification of the person given that they were swiped right.

If both processes is accepted then the contacts is built up and they are able to chat.



# SYSTEM DESIGN FLOW



# LINKED LIST

A sequence of nodes, where each node stores an element and a reference to the next node in the sequence.

Linked list in mitrata:

We have used a singly linear linked list to store user profiles temporarily within the user login session. It stores two user profiles as nodes.

If more than two nodes are used, the program becomes memory inefficient.  
Rather, we considered making two nodes:

1. for displaying first profile in the user feed .
2. for holding the data of next profile



# LSM TREE

LSM (Log-Structured Merge) tree is a data structure used to store and manage large amounts of data efficiently. LSM trees are commonly used in modern databases like Cassandra, HBase, and LevelDB.

Some of many advantages of LSM tree are:

- Fast writes - buffering new data in memory before flushing it to disk in large sequential writes
- Efficient storage - a tiered storage structure where data is stored in multiple levels, with the newest data in memory and older data on disk.
- Scalability - a tiered storage structure where data is stored in multiple levels, with the newest data in memory and older data on disk



# ELEMENTS OF WEB DEV.

Web development involves a combination of programming, design, and content creation to produce a functional and visually appealing website.

Web development can be divided into following main categories:

- Front-end development - the layout, design, and interactive elements  
the layout, design, and interactive elements, **react.js** is heavily used here.
- Back-end development - involves the server-side of, including the database, **Python** has been used here.
- Databases - We have stored data locally for now.
- Web servers - responsible for delivering web pages and content to clients that request them.



# OTHER DSA ELEMENTS USED

## Chat using stack:

stack is a data structure which works on the basis of LIFO

Stack can be used to store a chat conversation in a chat application. Each message sent and received in the conversation can be pushed onto the top of the stack, and when a message is deleted or popped from the stack, the most recent message is removed.

Stack can be used to store a chat conversation in a chat application. Each message sent and received in the conversation can be pushed onto the top of the stack, and when a message is deleted or popped from the stack, the most recent message is removed.



# FUTURE ENHANCEMENTS

There is always room for improvement, however small it may be.

*But in our case*, there is big room for improvement.

Due to the limited time constraint, we couldn't manage to add some features that we had planned.

some of the possible enhancements are:

- Replace the friend request feature with literal matching algorithm
- Implementing remote database
- Deploying the site
- Developing an mobile app for both IOS and Android

*Thank  
you!*



migrata