A Midterm Progress Report

On

PunjabVerse

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

(*Computer Science &Engineering)*

Manjot Kaur(2203497)
Jasvir Kaur(2203474)
Jasmin Kaur Gahlot(2203470)

UNDER THE GUIDANCE OF

Prof. Palak Sood

Department of Computer Science and Engineering

GURU NANAK DEV ENGINEERING COLLEGE,

LUDHIANA, 141013

INDEX

Abstract

Proverbs are deeply rooted in cultural contexts and often lose their meaning when translated literally. Existing translation tools fail to provide contextually accurate translations of Punjabi proverbs, making it difficult for users to understand their true essence. This project aims to develop a Flask-based proverb Translation System that enables users to search for Punjabi proverbs and receive contextual English translations along with equivalent English idioms. The system incorporates a searchable database that categorizes idioms based on themes such as wisdom, humor, and relationships. Additionally, it features a fuzzy search mechanism to accommodate minor spelling errors and variations, ensuring that users can find relevant idioms even if they do not enter the exact wording.

The system is structured into multiple modules, including a user-friendly interface for easy navigation, a search module for retrieving matching proverbs, a database module to store and manage proverbs. The project utilizes Flask (Python) for the backend, SQLite for database management, and HTML, CSS for the frontend.

By offering a structured and efficient way to translate Punjabi proverbs, this system addresses the limitations of existing translation tools and enhances language learning and cross-cultural understanding. It serves as a valuable resource for students, researchers, and language enthusiasts, promoting better accessibility to linguistic heritage.

Acknowledgment

We extend our sincere gratitude to Dr. Kiran Jyoti, Head of the Computer Science and Engineering Department at GNDEC, Ludhiana, for her constant guidance, encouragement, and invaluable insights throughout the development of our project PunjabVerse. Her unwavering support has been instrumental in shaping this project.

We are deeply indebted to our project guide, Prof. Palak Sood, for her expert supervision, patient mentorship, and constructive feedback at every stage of this work. Her technical expertise and dedication played a pivotal role in overcoming challenges and ensuring the successful completion of this project.

We also express our heartfelt appreciation all the faculty members of the Computer Science and Engineering Department, GNDEC, for their intellectual support, encouragement, and valuable suggestions. Their guidance helped refine the functionality and usability of the project.

Lastly, we acknowledge the contributions of everyone—friends, family, and peers—who supported us directly or indirectly during this journey. Their motivation and cooperation were vital in bringing this project to fruition

1. Introduction

1.1 Introduction: Language is a pillar of human culture, acting as a medium to express thoughts, share knowledge, and pass on traditions. Within every language lies a treasure trove of proverbs, concise and often metaphorical expressions that carry deep meanings, reflections, and collective wisdom. Punjabi, a vibrant and culturally rich language spoken by millions worldwide, boasts a vast collection of proverbs that offer insights into the customs, values, and philosophies of the Punjabi-speaking community. These proverbs, rooted in the daily lives and experiences of people, have been passed down through generations, preserving the essence of Punjabi heritage. However, with the growing emphasis on globalization and cross-cultural communication, the need to translate and interpret these proverbs for a wider audience has become increasingly significant.

Proverbs often present challenges during translation because their meanings are not always literal and are deeply tied to the cultural context in which they originate. A direct word-for-word translation frequently fails to convey the true intent, humor, or wisdom embedded in the proverb. This project addresses the gap between languages by developing a comprehensive system for translating Punjabi proverbs into English while maintaining accuracy, contextual relevance, and cultural depth.

The primary objective of this project is to compile a robust dataset of Punjabi proverbs, complete with their contextual English equivalents and English contextual meaning. By compiling a collection of proverbs, this project aims to preserve and digitize the cultural wisdom of the Punjabi language while making it accessible to a global audience.

To enhance usability, this project will incorporate a user-friendly interface designed for simplicity and functionality. The interface will allow users to easily input Punjabi proverbs in their original script and retrieve translations in English with proper formatting and readability. This design ensures that the system is inclusive and accessible, regardless of the user's linguistic background.

A critical aspect of this project is ensuring the linguistic accuracy of translations. Proverbs are not merely phrases but encapsulations of cultural and emotional experiences, and any inaccuracy in translation risks diminishing their essence.

This project is significant not only for its linguistic and cultural contributions but also for its potential to bridge gaps between languages and foster cross-cultural understanding. In today's interconnected world, effective communication and cultural appreciation are more important than ever. By providing a reliable resource for translating Punjabi proverbs into English, this project will enable people from diverse backgrounds to understand and appreciate Punjabi culture, thereby promoting cross-cultural dialogue and mutual respect.

Moreover, the project addresses the urgent need for digital tools that preserve endangered linguistic and cultural artifacts. As younger generations increasingly adopt dominant global languages like English, there is a risk of losing the richness of regional languages and their associated cultural elements, such as proverbs. This project will serve as a digital repository, safeguarding Punjabi proverbs for future generations while integrating them into the global digital landscape. By combining the traditional wisdom of Punjabi proverbs with modern technological solutions, the project ensures that this vital aspect of Punjabi culture remains relevant and accessible in the digital age.

In conclusion, this project is a multifaceted effort to preserve, digitize, and promote Punjabi proverbs while facilitating their accurate translation into English. It not only aims to provide a practical tool for translation but also seeks to celebrate and share the cultural richness of Punjabi heritage with a global audience. Through its comprehensive dataset, user-friendly interface, and commitment to linguistic accuracy, this project aspires to create a bridge between languages and cultures, enabling deeper appreciation and understanding of Punjabi proverbs in an increasingly interconnected world.

1.2 Objectives

Followings are the objectives of this Project:

1. To compile a comprehensive dataset of Punjabi proverbs with their contextual English meanings.
2. To create a user-friendly interface for translating Punjabi proverbs into English.
3. To ensure language accuracy, and interface functionality.

2. System Requirements

System requirements define the necessary hardware, software, and network resources for a project to function effectively. Given requirements ensure the system runs smoothly, meets performance expectations, and provides a good user experience

2.1 Hardware Requirements:

Hardware requirements specify the physical components needed to run the system efficiently. The hardware requirements provides the foundation for running software and application.

1. **For Development:** These are the basic hardware specifications required to develop and test the application.

   a. Processor: Intel Core i3

   b. RAM: Minimum 8 GB

   c. Storage: At least 20 GB free space

2. **For Deployment:** When deploying a web application, the hardware requirements depend on factors such as expected traffic, processing power, and storage needs. Below are different deployment options with corresponding hardware requirements.

   a. Processor: Multi-core processor

   b. RAM: Minimum 8 GB

   c. Storage: At least 50 GB (scalable for larger datasets)

   d. Network: High-speed internet connection

   e. Operating System: Linux or Windows Server

2.2 **Software Requirements**:

The software requirements specify the necessary software components needed for the development, deployment, and operation of the system. These requirements ensure the project runs efficiently across different environments.

1. **Development Tools:** Development tools are essential for writing, testing, and debugging code efficiently. Given are the development tools for proposed work.

   a. Programming Language: Python 3.9 or higher

   b. Operating system: Windows 10/11, Ubuntu 20.04+, macOS

2. **Frameworks and Libraries:** Frameworks and libraries provide essential functionalities to speed up development and ensure efficiency. Below is a list of the frameworks and libraries required for proposed work:

   a. Web framework: FLASK

   b. Database and object relational mapping: SQLite3,

   c. Frontend Technologies: HTML5, CSS

3. Software Requirement Analysis

3.1 **Introduction:** Software Requirement analysis is a foundational step in software development that focuses on identifying and documenting the objective of the project. It serves as the blueprint for the entire development process, ensuring that the final software product meets user expectations and business goals. A thorough and clear requirement analysis minimizes the risks of miscommunication, reduces costly changes later in the project, and sets the stage for efficient design and development.

3.2 **Problem Definition:**

Proverbs are a vital part of language and culture, often carrying meanings and wisdom that transcend their literal translations. Current translation tools, such as Google Translate, struggle with translation of proverbs due to their lack of contextual understanding and cultural sensitivity. As a result, literal translations often fail to convey the true essence of proverbs.

The need for a system that accurately translates proverbs while preserving their cultural significance is critical. This project aims to develop a Proverb Translation Website focused on translating Punjabi proverbs into their English contextual meaning. The system will ensure that the translations are not only accurate but also contextually and culturally meaningful.

Challenges in the exciting system:

1. Many translation tools provide word-for-word translations, which fail to capture the true meaning of Punjabi proverb.
2. The exciting system doesn’t focus on translation of the proverbs.
3. Finding English equivalent is not easy.

3.3 Proposed solution:

To address these challenges, the proposed work provides a system with accurate and reliable translation system that translate the Punjabi proverb in their English contextual meaning and English equivalent if exist.

Key features of the system:

1. **Contextual Meaning:** The excited system doesn’t provide the contextual meaning of proverbs but provide the literal translation. The proposed work provides the contextual meaning instead of literal translations.
2. **English equivalent of proverb:** Finding the English equivalent of any Punjabi proverb is not easy and time wasting process. The database of the project contains Punjabi proverbs with their closest English equivalents if exist.
3. **Centralized database:** The database of the project stores a structured collection of Punjabi proverbs with their contextual meaning and English equivalent if exist. In the dataset the data is organized and easily retrievable.
4. **Web-Based Interface:** The proposed work is a web application which translate with searching from database rather than using any AI, maintain the accuracy and humor of the language.

3.4 Module

A module is a separate, reusable part of a software system that performs a specific function. It helps organize the code and makes the system easier to manage, test, and update.

The Proposed work has three modules:

1. Proverb Dataset Management Module
2. Search Module

4. Frontend User Interface Module

3.5 Proverb Dataset Management Module

a. **Functionality:** The Proverb Database Management Module is used to store all proverbs with their contextual meaning in a structured format.

b. **Key features:** This module stores the Punjabi proverb with their contextual meaning and English equivalent. It allows easy retrieval of data for searching

c. **Implementation:** SQLite is used as database system.

d. **Work flow:**

   1. The database stores each proverb with an ID, their contextual meaning and English equivalent.
   2. When search module request a proverb it retrieve the correct data.
   3. The module sends the result back to the search module for processing.

3.6 Search Module:

a. **Functionality:** The search module processes the user input and finds the best matching proverb from the database.

b. **Key features:** This module finds the proverbs based on user input. This module provides the result even if there is spelling mistake.

c. **Implementation:** FLASK is used to implement this module.

d. **Work flow:**

   1. Search module takes user input from UI Module.
   2. It queries the database module for relevant result.
   3. It return a list of matching proverbs to the UI module.

3.7 Front end user interface module(UI module)

a. **Functionality:** The UI module provides an easy to use web interface for users to search, browse and view the translation.

b. **Key features:** UI module provides a search bar where user enter Punjabi proverb and display the search result with their English contextual meaning.

c. **Implementation:** HTML and CSS is used for user friendly interface.

d. **Work flow:**

1. User enter a Punjabi proverb in the search bar.
2. This module sends the input to the search module for processing.
3. It retrieve the result and display it on the webpage.

4. Software Design

4.1 Software design

Software design is a method that converts user requirements into a suitable form to employ in software coding and implementation.

To achieve a well-structured and efficient solution, different design models are used, including flowchart, UML diagrams, data flow diagram, sequence diagram, use case diagram. These diagrams illustrate the working of the system, module interaction and data processing, serving as a blueprint for the actual implementation. The following section provides detail insight into system design, covering its logical structure, functional components and deployment architecture.

4.2 System Architecture:

System architecture refers to the overall structure and design of a software system, including its components, modules, data flow, and interactions between different parts. It serves as a blueprint that defines how the system functions and how different components work together.

The proposed work follow a Three-Tier Architecture ensuring the balance between performance, scalability and security:

1. **Presentation Tier:**

   The presentation tier is the user interface and communication layer of the application, where the end user interacts with the application.

   In the Proposed work Presentation tier provides the search form to the user and provides the required output. It provides the responsive design that can run on desktop as well as mobile.

   Function: The main purpose of Presentation layer is to take input of Punjabi proverb from user and display English contextual meaning and English equivalent of searched Punjabi proverb.

Technology used: HTML and CSS is used to design the responsive and user-friendly interface in proposed work.

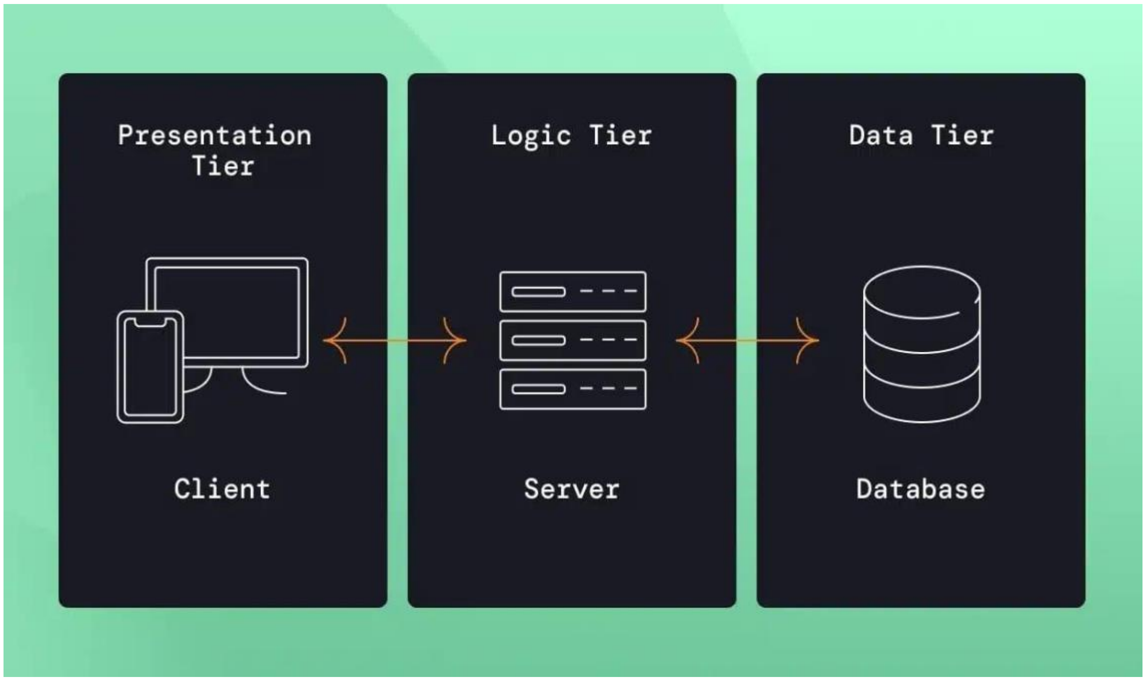


Figure 4.1 three tier architecture

2. **Logic Tier**

The application tier, also known as the logic tier or middle tier, is the heart of the application. In this tier, information that is collected in the presentation tier is processed. This tier interact with data tier to process the query of a user to provide the result.

In the proposed work Logic Tier takes the input as Punjabi proverb from presentation layer and interact with data tier to match the input with database entries to fetch the English contextual meaning of input Punjabi proverb.

Function: The main purpose of the Application layer is to controls system logic and database interactions.

Technology used: FLASK will be used to handle request and logic and flask SQLAlchemy will be used to connect FLASK to the database.

3. **Data Tier**

The data tier, sometimes called database tier, data access tier or back-end, is where the information that is processed by the application is stored and managed. The logic tier interact with data tier to search the data entries in database.

In the Proposed work Data Tier stores a structured Punjabi proverb data with their English contextual meaning and English equivalent if possible.

Function: The main purpose of the data tier is to stores the Punjabi proverb with their English contextual meaning and manage proverb translation.

Technology used: SQLite is used to store dataset of Punjabi proverbs and their English contextual meaning.

4.3 Flow Chart of Proposed Work

Flowchart is a diagrammatic representation of sequence of logical steps of a system. Flowcharts use simple geometric shapes to depict processes and arrows to show relationships and process or data flow.

The flow chart of the proposed work is designed to ensure fast searching and providing accurate English contextual meaning of the searched Punjabi proverb while maintaining the reliability of the language. Below is a step by step breakdown of the flow chart based on reference image:

1. **User input**

   a. The user enter the Punjabi proverb partially or fully in the search form.

2. **Flask receives the request**

   a. The input is sent to the FLASK backend using an HTTP request with POST or GET method.

b. The FLASK extracts the search query and prepare it for database processing.

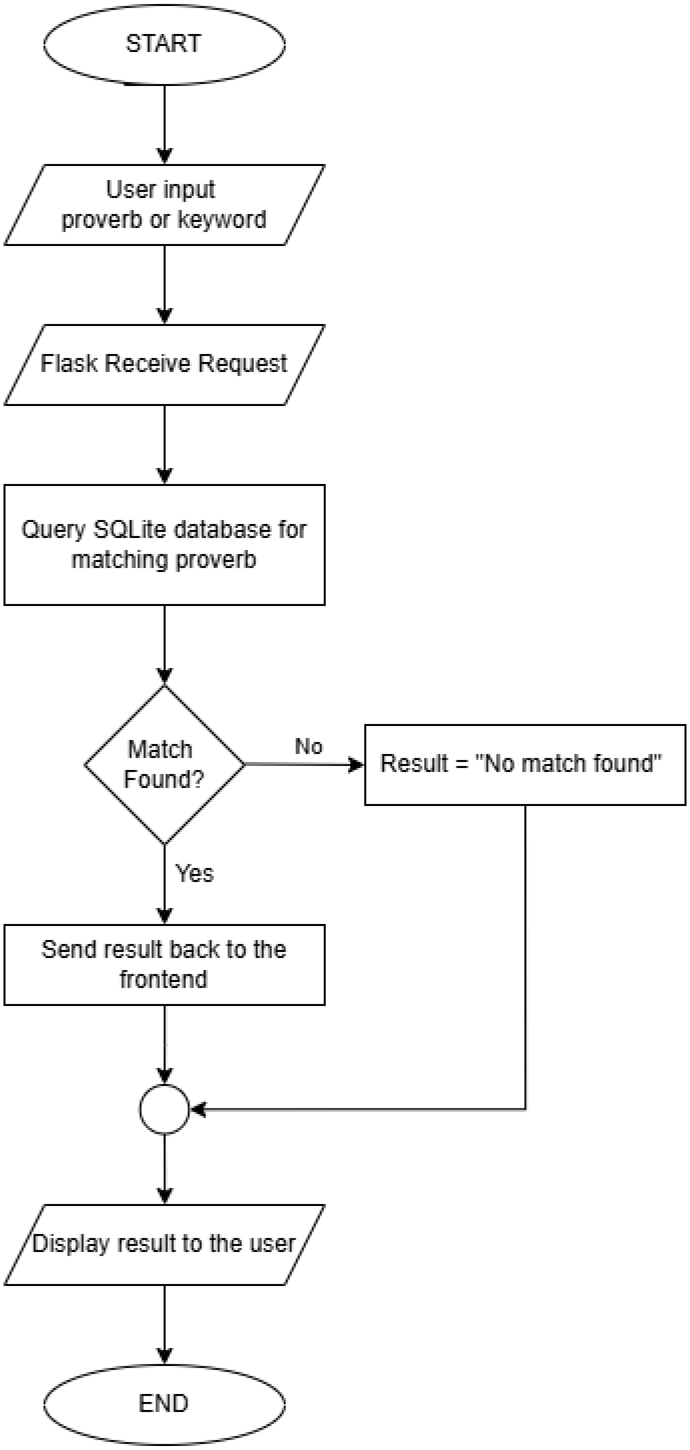

START
User input
proverb or keyword
Flask Receive Request
Query SQLite database for matching proverb
Match Found?
No
Result = "No match found"
Yes
Send result back to the frontend
Display result to the user
END

Figure 4.2 flow chart of proposed work

c. SQLite query

a. The system queries the SQLite database to find matching entries with the searched input.

b. The database contains Punjabi proverbs, their contextual meaning and English equivalent.

d. Matching the query

a. The system checks whether a matching proverbs exists in the database.

b. If the match found the system retrieves the proverb, its English equivalent and contextual meaning.

c. If match not found the system display a message for no matching proverb found.

e. Display Result

a. The retrieved proverb and its contextual meaning are send back to the frontend

b. The user sees the translated proverb on their screen.

4.4 Sequence Diagram of Proposed Work

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

A Sequence Diagram of Proposed Work shows the interaction between different components of the system.

4.4.1 The different Components in the sequence diagram:

Below are the different component of Sequence Diagram of Proposed work based on the reference image:

1. **User:** The user is the person that interact with the system to search the contextual meaning of Punjabi proverb.

2. **Web interface (front end):** Web interface is a HTML form where user enter the proverb and that interact with the backend to process the query.
3. **FLASK (back end):** The back end is used to process user request and communicate with the database.
4. **SQLite database:** The SQLite database is used to store and retrieve the proverb translation.

Sequence diagram of proposed work:

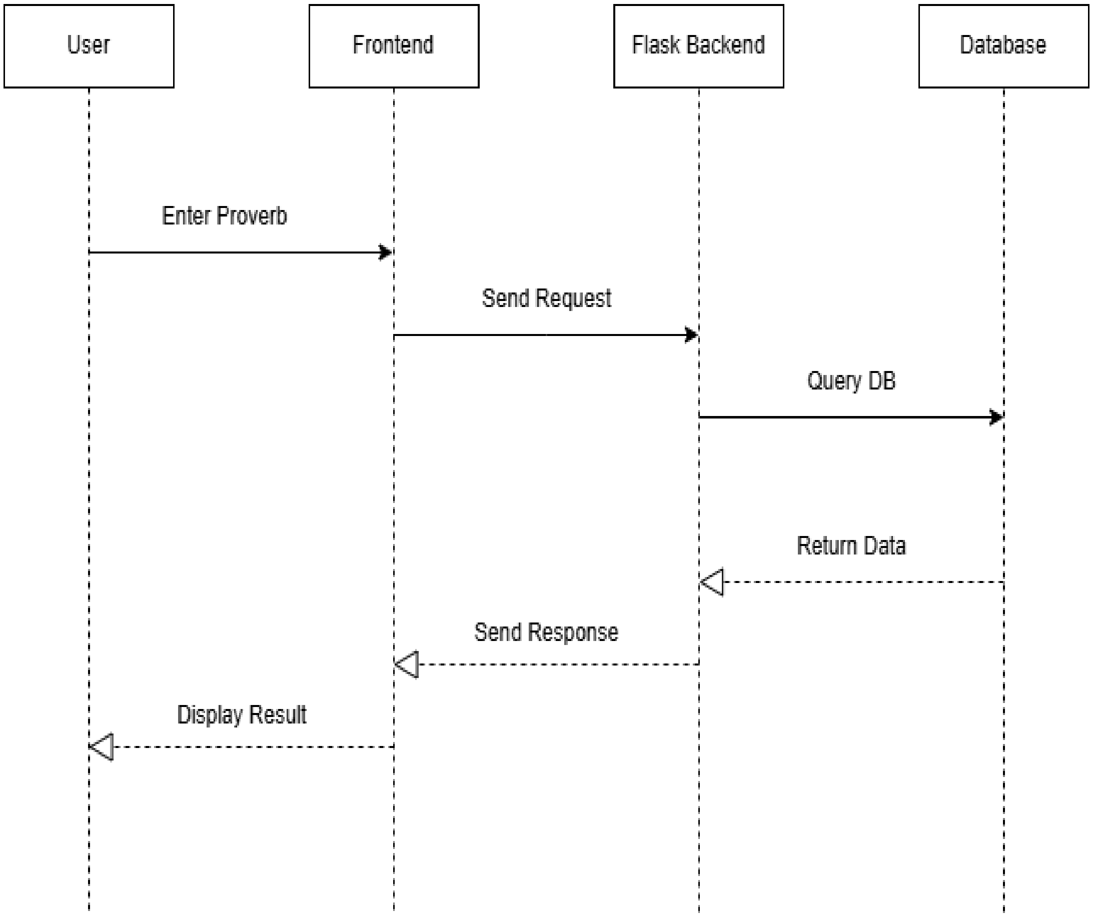


Figure 4.3 sequence diagram of Proposed work

4.4.2 Flow of sequence diagram:

Below given is the step by step breakdown of the sequence diagram of the proposed work based on the reference image:

1. **Enter the proverb:**
    a. User enter the Punjabi proverb as an input in the search form.
    b. User can enter complete proverb or can search partially also.
2. **Send request:**
    a. Front end receive the input from the user.
    b. Front end sends the request to the backend.
3. **Query Database:**
    a. FLASK backend receive the request of front end.
    b. The backend process the request and send the queries to the Database.
4. **Return data:**
    a. The database receive the queries from back end.
    b. Database search for a match with input.
    c. If database founds the match then it return the contextual meaning and equivalent of the Punjabi proverb.
    d. If database can't found the match then it return the message "no result found".
5. **Send response:**
    a. backend receive the response of the queries from the database.
    b. Backend sends the response back to the frontend.
6. **Display result:**
    a. Front end display the result of searching of the user.

5. Testing Module

5.1 **Testing Framework:** A testing framework is a group of guidelines that support testers in building test cases and performing software testing effectively. It also provides robust tools and methods to simplify the testing process. By using a testing framework, the accuracy and performance of software testing can be increase to high levels.

5.2 Test Cases:

A test case is a set of actions performed on a system to determine if it satisfies software requirements and functions correctly. The purpose of a test case is to determine if different features within a system are performing as expected and to confirm that the system satisfies all related standards, guidelines and customer requirements. The process of writing a test case can also help reveal errors or defects within the system. A sequence or collection of test cases is called a test suite.

5.3 System Testing for proposed work:

1. Functional Testing:

Functional Testing is used to test the core features of the system. Its goal is to ensure the search and translation of proposed work correctly. Given are the test cases that will use for functional testing:

1. Valid proverb search: Enter all the Punjabi proverbs that are in the dataset to ensure whether system provides correct English contextual translation.
2. Invalid proverb search: Enter any random text to ensure that system should return no result found as output.
3. Case Insensitivity Test: Enter the proverb whether using phonetics or inscript Punjabi typing the search should return the same result regardless of input case.

4. Empty input handling: Click the search button without entering any text. The system should show an error message prompting the user to enter a search term.

2. **UI/UX Testing:**

   UI/UX Testing is used to test the responsiveness of the user interface. Its goal is to ensure the website works smoothly across devices. Below are some test cases used for UI/UX testing:

   1. Mobile compatibility: Try to open the website on a mobile browser. The layout should be responsive and work correctly.
   2. Button and navigation testing: Try to click all the button. All button should function correctly and not crash the site.

3. **Performance Testing:**

   The Performance testing is used to check the speed and load handling capability of the system. Its goal is to ensure the system can handle multiple searches efficiently. Below are some test cases used for performance testing.

   1. Response time test: Measure the time taken for a search request to return result. Response time should be fast

   2. Load testing: simulate multiple users searching at the same time. The system should handle multiple users without crashing.

6. Performance of Project Development

The PunjabVerse project has made significant progress. The database of Punjabi proverbs and the user friendly interface is developed for the system. The following are the key accomplishment that are done so far.

6.1 Key accomplishment:

1. **Database Creation**:

   The database is a crucial part of the proposed work as it stores and manage Punjabi proverbs, English contextual meaning and if possible then their English equivalent. Key features of database is given below:

   a. A structured SQLite database of 200 proverbs is designed to store Punjabi proverbs and their contextual English translations.

   b. The database included Punjabi proverbs, their English contextual meaning and English equivalent for some proverbs.

   c. The database is consistent and contain error free translation. We try to prevent the duplicate entries and incorrect or inaccurate contextual meaning of Punjabi proverbs.

2. **Frontend Development**

   The front end is the user facing part of PunjabVerse. It is what user interact with when they search for Punjabi proverbs and view translation. A well designed front end is essential for usability, accessibility and user engagement. Some key feature of designed front end of the proposed work are given below:

   a. A user-friendly interface is built for the website using HTML and CSS.

   b. The front end Provide a Search functionality for users to input proverbs.

c. Support a responsive design for optimal usability across different devices like desktop, mobile, tablet.

d. Focused on creating an intuitive and visually appealing layout to enhance the overall user experience.

The project has achieved approximately 65-70 % completion.

6.2 Next step to build the accomplishment:

1. **Backend Implementation**:

The backend is the core engine of PunjabVerse. It handle data processing, database interactions. Without a strong backend the system won't function properly. Some of the features of backend that we will try to implement are given below:

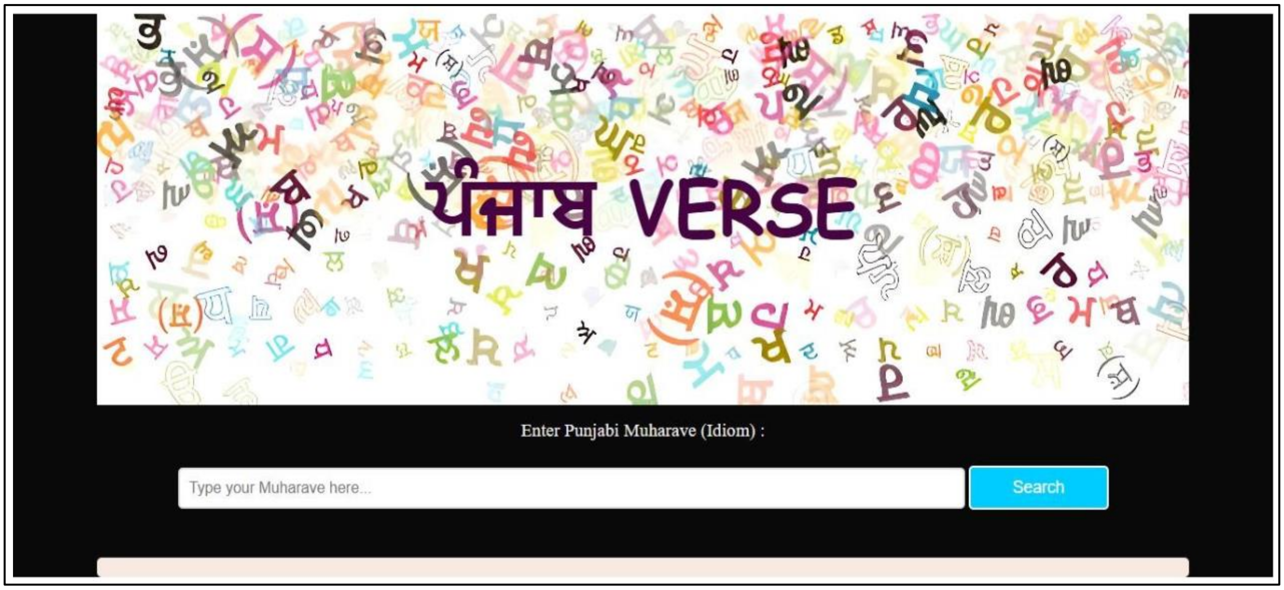
a. Develop a Flask-based backend to handle user inputs, query the SQLite database, and return translation.

b. Support case insensitive and partial word searches. Try to use fuzzy matching to find results even if the spelling is incorrect.

c. Website deployment so that user can access the website from anywhere.

7. Output screens

Front end of proposed work:

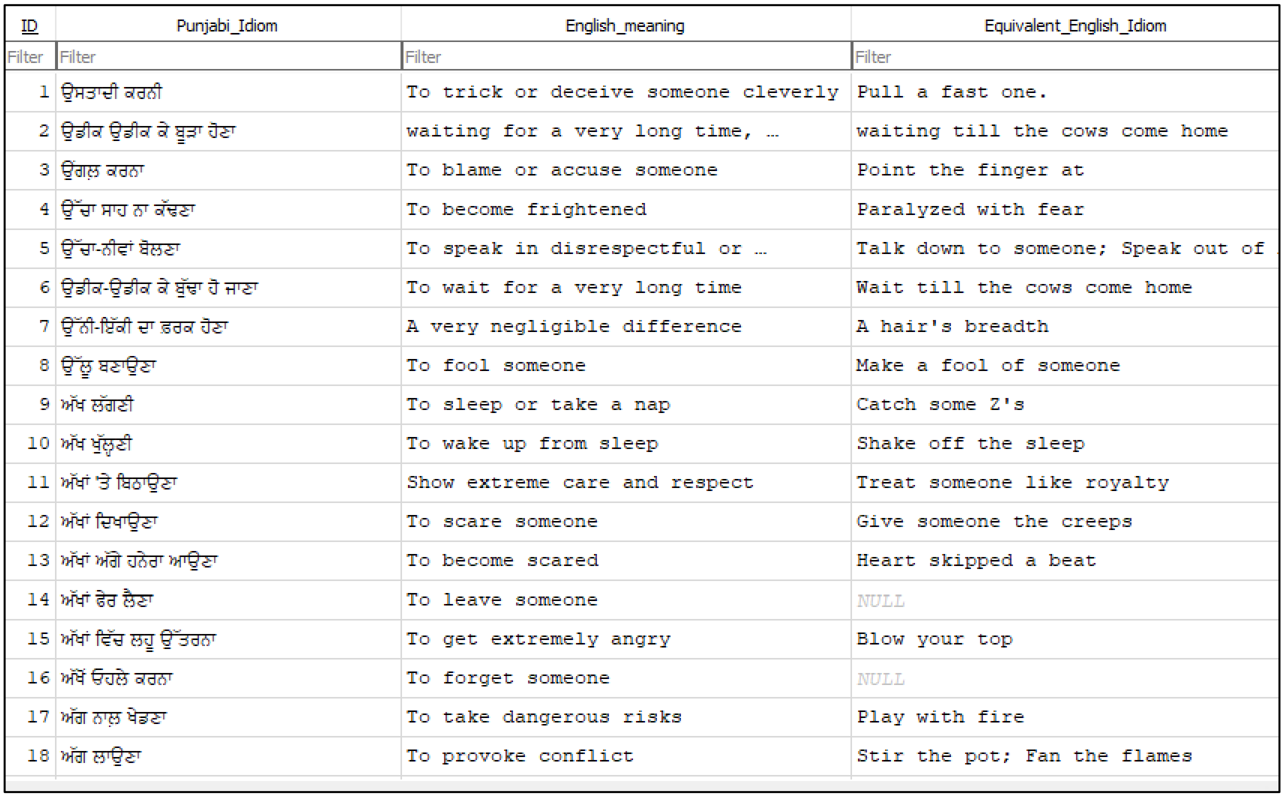


Enter Punjabi Muharave (Idiom) :

Type your Muharave here...

Figure 7.1 front end of the proposed work

Database of proposed work:



ID	Punjabi_Idiom	English_meaning	Equivalent_English_Idiom
Filter	Filter	Filter	Filter
1	ਉਸਤਾਦੀ ਕਰਨੀ	To trick or deceive someone cleverly	Pull a fast one.
2	ਉਡੀਕ ਉਡੀਕ ਕੇ ਬੂੜਾ ਹੋਣਾ	waiting for a very long time, …	waiting till the cows come home
3	ਉਂਗਲ਼ ਕਰਨਾ	To blame or accuse someone	Point the finger at
4	ਉੱਚਾ ਸਾਹ ਨਾ ਕੱਢਣਾ	To become frightened	Paralyzed with fear
5	ਉੱਚਾ-ਨੀਵਾਂ ਬੋਲਣਾ	To speak in disrespectful or …	Talk down to someone; Speak out of
6	ਉਡੀਕ-ਉਡੀਕ ਕੇ ਬੁੱਢਾ ਹੋ ਜਾਣਾ	To wait for a very long time	Wait till the cows come home
7	ਉੱਨੀ-ਇੱਕੀ ਦਾ ਫ਼ਰਕ ਹੋਣਾ	A very negligible difference	A hair's breadth
8	ਉੱਲੂ ਬਣਾਉਣਾ	To fool someone	Make a fool of someone
9	ਅੱਖ ਲੱਗਣੀ	To sleep or take a nap	Catch some Z's
10	ਅੱਖ ਖੁੱਲ੍ਹਣੀ	To wake up from sleep	Shake off the sleep
11	ਅੱਖਾਂ 'ਤੇ ਬਿਠਾਉਣਾ	Show extreme care and respect	Treat someone like royalty
12	ਅੱਖਾਂ ਦਿਖਾਉਣਾ	To scare someone	Give someone the creeps
13	ਅੱਖਾਂ ਅੱਗੇ ਹਨੇਰਾ ਆਉਣਾ	To become scared	Heart skipped a beat
14	ਅੱਖਾਂ ਫੇਰ ਲੈਣਾ	To leave someone	NULL
15	ਅੱਖਾਂ ਵਿੱਚ ਲਹੂ ਉੱਤਰਨਾ	To get extremely angry	Blow your top
16	ਅੱਖੋਂ ਓਹਲੇ ਕਰਨਾ	To forget someone	NULL
17	ਅੱਗ ਨਾਲ਼ ਖੇਡਣਾ	To take dangerous risks	Play with fire
18	ਅੱਗ ਲਾਉਣਾ	To provoke conflict	Stir the pot; Fan the flames

Figure 7.2 Database

8. References

[1] A. Punjabi, “9th-10th Adhunik Punjabi vyakran ate lekh Rachna-9.pdf,” *Google Docs*, 2019.https://drive.google.com/open?id=1wXHjkmJQwxsiXjzHSNdbDfKJaHVJdw96&authuser=punjabeducare%40ppppjalandhar.org&usp=drive_fs (accessed Apr. 03, 2025).

[2] I. Flat, “Colorful vector background made from Punjabi Gurmukhi alphabets,...,” *iStock*, Jun. 21, 2022. https://www.istockphoto.com/vector/colorful- vector-background-made-from-punjabi-gurmukhi-alphabets-scripts-letters-or- gm1404145148-456449715?searchscope=image%2Cfilm (accessed Apr. 03, 2025).

[3] S. A. Hamdi, S. E. Sa, Kazuhiko Nakae, Nakae@kansaigaidai Ac, and Okasha, “Online Translation of Proverbs between availability and accuracy,” 2013 International Symposium on Language, Linguistics, Literature and Education, Nov.2013, Available: https://www.researchgate.net/publication/297778929_ONLINE_TRANSLATION_OF_PROVERBS_BETWEEN_AVAILABILITY_AND_ACCURACY

[4] Joana Kondo Ismaili, “Problematic Areas in the Translation of Proverbs,” Journal of Cultural and Religious Studies, vol. 6, no. 11, Nov. 2018, doi: https://doi.org/10.17265/2328-2177/2018.11.003