**Hackathon Submission Template (Level-2-Solution)**

**Use Case Title:**

**Blind-Friendly Coding Platform**

**Student Name: Aishwarya .T.P**

**Register Number: C2S29102**

**Institution: Madurai Gandhi N.M.R.Subbaraman College for Women**

**Department: B.Sc (Computer science)**

**Date of Submission: 03.04.2025**

**1. Problem Statement**

For visually impaired individuals, coding presents significant challenges. Many screen readers struggle with interpreting complex code structures, making it difficult for blind developers to navigate, debug, and write efficient programs. Debugging becomes even harder due to the inability to visually scan error messages quickly. Furthermore, modern UI frameworks often rely on graphical interfaces that are not optimized for accessibility, limiting the participation of visually impaired programmers in software development.

### 2. ****Proposed Solution****

To address these challenges, we propose **AccessibleCodingDB**, a platform that enables blind developers to code efficiently with better accessibility features. The solution includes:

* A database-driven system to store and retrieve code snippets in an accessible format.
* Integration with screen readers and text-to-speech (TTS) technology for reading code aloud.
* An error log and debugging assistance module to provide structured debugging steps.
* A mentorship system connecting visually impaired developers with experienced mentors.
* Accessibility tools repository to help developers choose the best screen readers and debugging tools.

By implementing these features, the platform will empower visually impaired programmers to work more effectively and independently.

## ****2. Database Design & Implementation****

### ****2.1 Database Creation & Tables****

CREATE DATABASE IF NOT EXISTS BlindFriendlyCodingDB;

USE BlindFriendlyCodingDB;

-- Users Table

CREATE TABLE IF NOT EXISTS Users (

UserID INT PRIMARY KEY AUTO\_INCREMENT,

FullName VARCHAR(255) NOT NULL,

Email VARCHAR(255) UNIQUE NOT NULL,

PasswordHash VARCHAR(255) NOT NULL,

Role ENUM('Developer', 'Mentor', 'Admin') DEFAULT 'Developer'

);

-- Code Snippets Table

CREATE TABLE IF NOT EXISTS CodeSnippets (

SnippetID INT PRIMARY KEY AUTO\_INCREMENT,

UserID INT,

Title VARCHAR(255) NOT NULL,

ProgrammingLanguage VARCHAR(50),

Code TEXT NOT NULL,

CreatedAt TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (UserID) REFERENCES Users(UserID) ON DELETE CASCADE

);

-

- Accessibility Tools Table

CREATE TABLE IF NOT EXISTS AccessibilityTools (

ToolID INT PRIMARY KEY AUTO\_INCREMENT,

Name VARCHAR(255) NOT NULL,

Description TEXT,

Compatibility VARCHAR(255) NOT NULL -- e.g., 'JAWS, NVDA, VoiceOver'

);

-- Debugging Support Table

CREATE TABLE IF NOT EXISTS DebuggingLogs (

LogID INT PRIMARY KEY AUTO\_INCREMENT,

UserID INT,

ErrorMessage TEXT NOT NULL,

DebuggingSteps TEXT,

Timestamp TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (UserID) REFERENCES Users(UserID) ON DELETE CASCADE

);

-- Mentorship Table (Connecting Blind Developers with Mentors)

CREATE TABLE IF NOT EXISTS Mentorship (

MentorshipID INT PRIMARY KEY AUTO\_INCREMENT,

MentorID INT,

DeveloperID INT,

StartDate DATE,

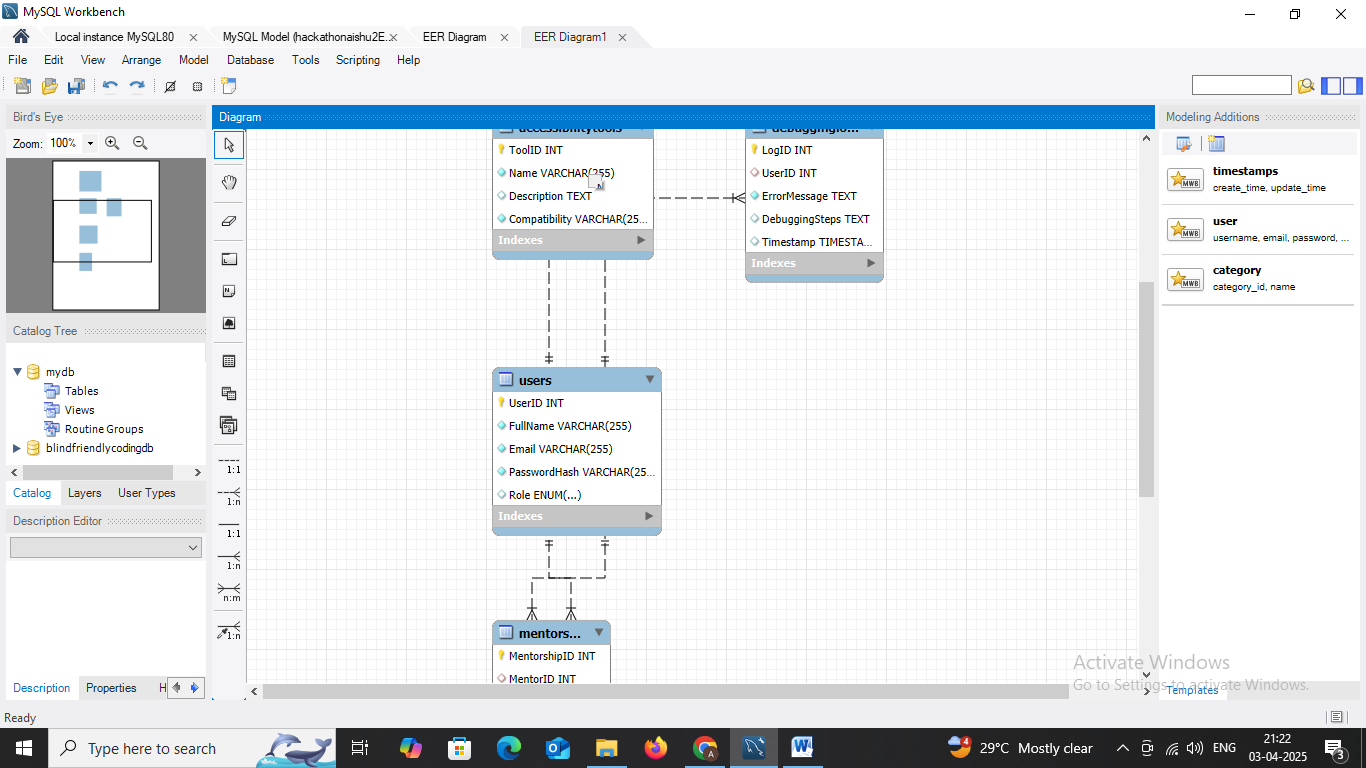
Status ENUM('Active', 'Completed', 'Pending') DEFAULT 'Pending',

FOREIGN KEY (MentorID) REFERENCES Users(UserID) ON DELETE CASCADE,

FOREIGN KEY (DeveloperID) REFERENCES Users(UserID) ON DELETE CASCADE

);

### ****2.2 ER Diagram (Reverse Engineered)****



## ****3. Queries for Data Management****

### ****3.1 Insert Sample Data****

-- Insert sample users

INSERT INTO Users (FullName, Email, PasswordHash, Role) VALUES

('John Doe', 'john@example.com', 'hashedpassword1', 'Developer'),

('Alice Smith', 'alice@example.com', 'hashedpassword2', 'Mentor'),

('Bob Admin', 'bob@example.com', 'hashedpassword3', 'Admin')

AS newUser

ON DUPLICATE KEY UPDATE Email = newUser.Email;

-- Insert sample code snippets

INSERT INTO CodeSnippets (UserID, Title, ProgrammingLanguage, Code) VALUES

(1, 'Hello World in Python', 'Python', 'print("Hello, World!")'),

(1, 'Basic HTML Structure', 'HTML', '<!DOCTYPE html><html><head><title>Test</title></head><body></body></html>')

AS newSnippet

ON DUPLICATE KEY UPDATE Title = newSnippet.Title;

-- Insert sample accessibility tools

INSERT INTO AccessibilityTools (Name, Description, Compatibility) VALUES

('JAWS', 'Screen reader for visually impaired users.', 'Windows'),

('NVDA', 'Free and open-source screen reader.', 'Windows'),

('VoiceOver', 'Built-in screen reader for macOS.', 'macOS')

AS newTool

ON DUPLICATE KEY UPDATE Name = newTool.Name;

-- Insert sample debugging logs

INSERT INTO DebuggingLogs (UserID, ErrorMessage, DebuggingSteps) VALUES

(1, 'SyntaxError: Unexpected token', 'Checked for missing semicolons and unmatched brackets.'),

(1, 'ModuleNotFoundError: No module named numpy', 'Installed the missing package using pip install numpy.')

AS newLog

ON DUPLICATE KEY UPDATE ErrorMessage = newLog.ErrorMessage;

-- Insert sample mentorship data

INSERT INTO Mentorship (MentorID, DeveloperID, StartDate, Status) VALUES

(2, 1, '2025-04-01', 'Active')

AS newMentorship

ON DUPLICATE KEY UPDATE Status = newMentorship.Status;

### ****3.2 Retrieval Queries****

-- Retrieve all users

SELECT \* FROM users;

-- Retrieve all code snippets by a particular developer

SELECT \* FROM codesnippets WHERE UserID = 1;

-- Retrieve all debugging logs for a particular developer

SELECT \* FROM debugginglogs WHERE UserID = 1;

-- Retrieve active mentorships

SELECT \* FROM mentorship WHERE Status = 'Active';

## ****4. Implementation & Results****

### ****4.1 Execution Environment****

This project was implemented and executed using **MySQL Workbench 8.0**.

### ****4.2 Screenshots of Execution Results****

### 

## ****5. GitHub Repository****

### ****5.1 Repository Link****

<https://github.com/A09I/hackathonaishu2>

### ****5.2 Uploaded Files in Repository****

* SQL database creation script
* ER diagram
* Sample data insertion script
* Retrieval queries
* Screenshots of execution results
* Documentation