ST. TERESA'S COLLEGE (AUTONOMOUS), ERNAKULAM

AFFILIATED TO MAHATMA GANDHI UNIVERSITY

PROJECT REPORT ON

YOUKAN HEAL DASHBOARD APP

In partial fulfilment of the requirements for the award of the degree of

BACHELOR OF SCIENCE IN
COMPUTER APPLICATIONS [TRIPLE MAIN]

Submitted By

FEMINA MERIN ROBY

III BSc Computer Applications [Triple Main]

Register No: SB22CA016

Under the Guidance of
Assistant Professor
Ms. Raji S Pillai

DEPARTMENT OF COMPUTER APPLICATIONS

2022-2025

ST. TERESA'S COLLEGE
(AUTONOMOUS)
ESTD 1925

CERTIFICATE

This is to certify that the project entitled **"YOUKAN HEAL DASHBOARD APP"** , is a bona-fide record work done by **FEMINA MERIN ROBY** [Reg. No. SB22CA016] during the year 2022-2025 and submitted in partial fulfilment of the requirements of the degree of Bachelor of Science in Computer Applications (Triple Main) under Mahatma Gandhi University.

17/03/2025
Head of the Department

Internal Examiner

ST. TERESA'S COLLEGE
DEPT. OF COMPUTERS
ERNAKULAM

17/3/25
External Examiner

Date: 14/03/25

DECLARATION

I, FEMINA MERIN ROBY (Register no: SB22CA016), B.Sc. Computer Applications [Triple Main] final year student of St. Teresa's College (Autonomous), Ernakulam, hereby declare that the project submitted named "YOUKAN HEAL DASHBOARD APP" for the Bachelor's Degree in Computer Applications [Triple Main] is my original work. I further declare that the said work has not previously been submitted to any other university or academic body.

Date: 17/03/2025

Place: ERNAKULAM

FEMINA MERIN ROBY

ACKNOWLEDGEMENT

Firstly, I thank God, the Almighty, with whose blessings, this project has been successfully completed.

I would like to convey my heartfelt gratitude to Manager Rev. Sr. Nilima (CSST), Directors Rev. Sr. Tessa (CSST), Rev. Sr. Francis Ann (CSST) and Principal Dr. Alphonsa Vijaya Joseph for providing me with this wonderful opportunity to work on a project with the topic YouKan Heal Dashboard App.

I would like to express my profound gratitude to the Head of the Department of Computer Applications Ms. Sheeba Emmanuel, my project guide Ms. Raji S Pillai and all other faculty of the department for their contributions to the completion of my project. The completion of the project would not have been possible without their help and insights.

Finally, I take this opportunity to thank all of them who has directly or indirectly helped me with my project.

FEMINA MERIN ROBY

SYNOPSIS

The YouKan Heal Dashboard App is an application created as a part of the YouKan Heal initiative, a collaborative program by Kochi Municipal Corporation, GIZ, C-HED, and St. Teresa's College (Autonomous). The initiative already has a web dashboard, and this project takes its functionalities to an app to enhance participation and make it more accessible and user-friendly. The app tracks engagement in three environmental challenges: Ban the Bag, Sustainable Menstruation, and Green Protocol. It allows students and institutions to submit their participation details, which are then analysed and visualized through an interactive dashboard.

Built using Flutter and Firebase, the application provides real-time statistics, graphs, and trends to measure the collective impact of these initiatives. With this application users can log in, participate in the challenges by filling out forms, and see their contribution to environmental change on the dashboard page in real time. Admins have extra features, including viewing challenge-specific data and institution management.

Through its user-friendly platform, the YouKan Heal Dashboard App promotes more active participation in green activities. It helps in keeping institutions and individuals connected, promoting a feeling of responsibility toward environmental preservation and collective effort.

CONTENTS

Contents

1. INTRODUCTION

1.1 About the Project

The YouKan Heal Dashboard App is a mobile app extension of Youkan Heal initiative, which is a collaborative venture of Kochi Municipal Corporation, GIZ, C-HED, and St. Teresa's College. The app facilitates students and institutions to engage in three sustainability challenges— Ban the Bag, Sustainable Menstruation, and Green Protocol—by entering data using an intuitive platform. Developed using Flutter and Firebase, this application allows real-time monitoring of engagement, interactive visualization of data in the form of line charts and pie charts, and management features for institutions and validation of entries.

1.2 Objectives of the Project

- Provide a mobile-friendly platform for easy participation in sustainability challenges.
- Enable real-time data tracking and interactive visualization.
- Offer institution and data management features for administrators.
- Encourage wider engagement in environmental initiatives.

2. SYSTEM ANALYSIS

2.1 Introduction

The primary function of system analysis is to examine existing processes and identify inefficiencies, and develop plans for optimizing system performance. Data collection and analysis, the identification of user needs, and the assessment of technological feasibility are required to develop an efficient and well-structured solution. That includes identifying possible problems and the steps to deal with them.

System analysis supports a fluent design of workflow, good data management, and integration of components with minimum disruption. System analysis promotes high usability, high scalability, and flexibility and minimizes errors and redundancies. The systematic analysis of the system supports better decision-making to enhance its reliability, efficiency, and effectiveness as a whole.

2.2 Existing System

The current system is a website displaying numbers related to various campaign initiatives. Data entry is a task assigned to the admin who inputs data collected from the participants through Google Forms. Such an approach is an inefficient answer: it relies heavily on traditional data management systems that are tedious and prone to error. Without an automated feature, retrieving and aggregating data for analysis becomes a very challenging task. The system in use is also incapable of proper data visualization, which is quite debilitating, owing to analysing trends and deriving insights. The absence of integrated data collection, entry, and rendering solutions very much restricts the efficiency and scalability of the system. Managing campaign data in the absence of a centralized and dynamic approach is tedious and results in significant loss in terms of valuable information processing time and efficiency.

2.3 Proposed System

The proposed system is an app for the mobile phone to solve the inefficiencies of the present process using a more systematic, accurate, and efficient method for collecting, processing, and presenting information. It guarantees that information is being collected systematically to avoid errors and the time lag associated with manual typing. It allows simpler access and better management of the records.

One of the main improvements is in data entry and updation. Instead of manual entry of the data by the admin, the system allows for direct submission of participation details by participants themselves, thus making data collection better organized and reliable. This reduces chances of discrepancy and ensures all records are up to date.

Additionally, instead of the presentation of just numbers, it provides a clear and organized representation of information as graphs and dashboards, allowing for better understanding and decision-making. Through transmission of significant insights on participation behaviours and trends, it is feasible to track effectiveness of efforts as well as places in need of improvement.

Another significant feature of the system is that it can process information efficiently when the number of initiatives increases. In contrast to the current method, which is inefficient at processing bulk data, the new system has all the information kept properly organized and readily accessible, irrespective of how many participants or initiatives there are.

Moreover, the system is designed more user-friendly, with a neater experience for administrators and participants in handling the information. The absence of unnecessary complexity allows administrators to administer and keep records easily, while participants get to experience a smooth and easy submission experience.

In general, the proposed system solves the issues of the existing method by improving the way data is collected, processed, and displayed. It is more accurate, effective, and accessible, with easy tracking of progress and informed decision-making. With an improved and more reliable system, the process is simpler, more effective, and more suitable for supporting the goals of various initiatives.

2.4 System Specifications

An effective solution is provided for collecting, managing and visualizing data by the system. It enables real-time updates with a structured data entry system, which provides user-friendly interaction between administrators and participants. The entire process of maintaining the information streamlines for accuracy, consistency, and easy accessibility. It was designed to scale, so it could run regardless of the number of users or initiatives. A user-friendly interface is designed for easing navigation and improving how systems operate with the users. Specifications deal with reliability, efficiency and accessibility, so that the system will fulfil the requirements of effective information management and decision making in organizations.

2.5 Operating System

The system requires Android Operating System for its smooth and proper execution. Android is a very flexible and well-supported mobile platform to ensure it runs on many mobile devices. It also provides a simple interface, effective resource management, and constantly seamless connection for applications to run smoothly. The system is designed so that it can maximize the performance of any Android device. Thus, a reliable, scalable, and easy-to-use system for both the administrator and the participant. Furthermore, this design ensures that the system is updated frequently and that security enhancements are made, making sure that it remains stable and high-performing even in the years to come.

2.6 Languages or Software Packages

- **Flutter:** Google-developed open-source technology enables the setup of natively compiled apps for mobile, web, and desktop use all from a single codebase. It presents a good development environment thanks to the extensive widgets' provision, allowing for pleasant user experiences. The Flutter reload ability is an important step in revitalizing productivity as it lets developers play around a design change in real time without needing to re-login.

- **Dart:** Language in which Flutter is created. It was devised to develop mobile and web applications from the ground upwards abiding by restrictions of performance and simplicity. Dart works with the effect of being a "dart" that works fast on the desired target as it empowers JIT compilation for faster development cycles and AOT-compilation for optimized performance for production.

- **Firebase Realtime Database:** Firebase Realtime Database is a cloud-hosted NoSQL database which allows storing and syncing of data in real-time all across various devices. It offers efficient data management, local support, and automatic updates, making it an ideal database to serve applications that necessitate instant data synchronization and scaling ability.

- **Android Studio:** This is the official Integrated Development Environment (IDE) for developing applications on Android. It has provided tools for advanced code editing, debugging, and performance analysis, not mentioning the emulator for testing applications on multiple devices. Android Studio supports Flutter development by integrating Flutter inbuilt and Dart plugins, thus streamlining development.

- **Firebase Authentication:** This method allows Firebase Authentication to operate and authenticate users using email/password. It further simplifies the identification process, enhancing the user experience in ensuring fast, secure login, without the user dealing with an added complexity of handling authentication on their own.

2.7 Hardware and Software Specifications

Hardware Requirements:

- Main Processor: Quad-core processor (Intel Core i3 or above)
- RAM: 8 GB or above
- Storage: 10 GB or more of free space
- Drive: 256 GB SSD or above (for smooth performance)

- Display: Minimum 1366×768 resolution

Software Requirements:

- Operating System: Windows 10/11 (64-bit) or macOS/Linux for development, Android for deployment
- Programming Language: Dart
- Development Framework: Flutter
- Database Management System: Firebase Realtime Database
- Development Tool: Android Studio or Visual Studio Code

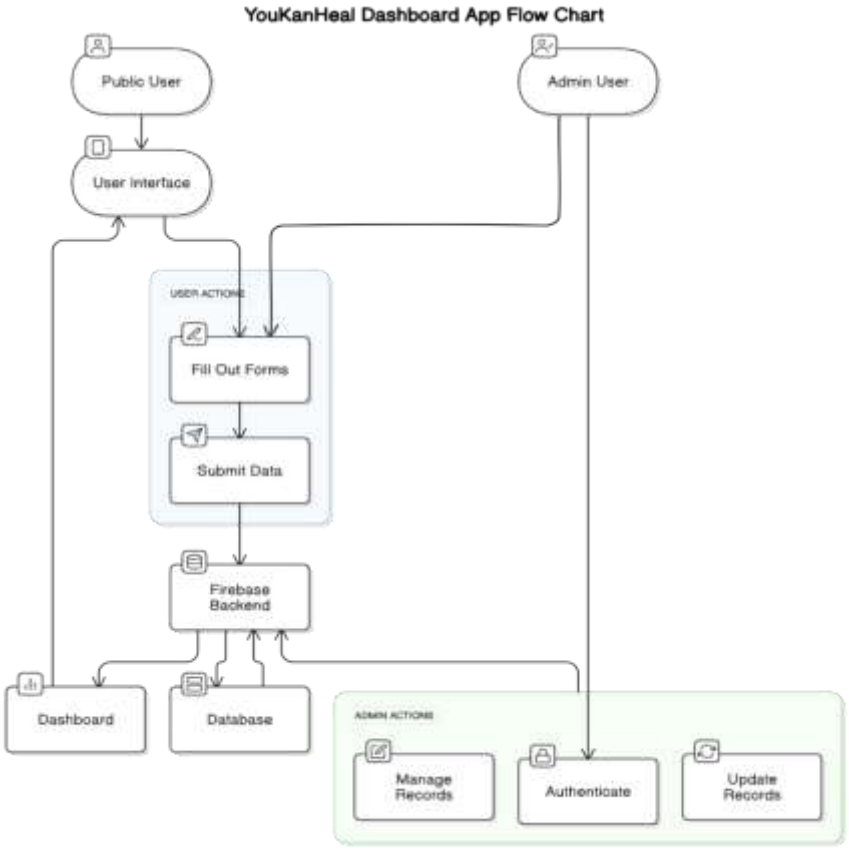
3. SYSTEM DESIGN

3.1 Introduction

System design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. It is a critical step in the software development lifecycle that bridges the gap between the requirements analysis and implementation phases.

3.2 Data Flow Diagram

A data flow diagram (DFD) is a graphical representation of the flow of data through an information system. A DFD is often used as a primary step to create an overview of the system, which can later be elaborated. A DFD shows what will be the input of the system as well as the output. It clearly represents where the data will come from and go to and where the data will be stored.



3.3 Data Dictionary

A data dictionary is a collection of metadata that provides information about the data used in a database or information system. It serves as a reference guide for data elements and their definitions, as well as their relationships with other data elements in the system. A data dictionary is useful for ensuring data accuracy, consistency, and completeness. It can also help facilitate communication and collaboration among stakeholders who are involved in designing, developing, and maintaining a database or information system.

3.4 Data Design

Database design is the organization of data according to a database model. A well designed database ensures data accuracy, consistency, and completeness and provides an intuitive and user-friendly interface for interacting with the data. Data design is the first design activity, which results in less complex, modular, and efficient program structure. The information domain model developed during the analysis phase is transformed into data structures needed for implementing the software. The data objects, attributes, and relationships depicted in entity relationship diagrams and the information stored in data dictionary provide a base for data design activity.

ban_the_bag_form

Field	Type	Description
entry_id	varchar	Unique identifier, autogenerated
carry_own_bag	boolean	Not Null
category	enum	{School, UG, PG, Faculty}

institution	varchar	Foreign key, referencing institutions
name	varchar	Not Null
non_biodegradable_bags	int	Not Null
preference_reason	enum	{Cost, Availability, Durability, Eco-friendliness, Other}
reusable_bag_material	enum	{Cloth, Jute, Paper, Plastic, Other}
timestamp	timestamp	Not Null
willing_to_participate	boolean	Not Null

green_protocol_form

Field	Type	Description
entry_id	varchar	Unique identifier, autogenerated
department_name	varchar	Not Null
event_date	date	Not Null
event_name	varchar	Not Null
institution	varchar	Foreign key, referencing institutions
institution_type	enum	{School, College}
items_used	array	List of selected items
number_of_participants	int	Not Null
volunteer_name	varchar	Not Null

institutions

Field	Type	Description
entry_id	varchar	Primary key

institution	varchar	Not Null

sustainable_menstruation_form

Field	Type	Description
entry_id	varchar	Unique identifier, autogenerated
category	enum	{School, UG, PG, Faculty}
disposable_products	int	Not Null
institution	varchar	Foreign key, referencing institutions
name	varchar	Not Null
timestamp	timestamp	Not Null
uses_cloth_pad	boolean	Not Null
uses_menstrual_cup	boolean	Not Null
uses_plastic_pads	boolean	Not Null
willing_to_participate	boolean	Not Null

4. SYSTEM DEVELOPMENT

4.1 Introduction

System development is an act of planning, designing, implementing, testing, and maintaining a software application. This organized approach is called the System Development Life Cycle (SDLC), which guarantees the proper development and management of the system. Different models are used in SDLC, such as the waterfall model, iterative, incremental, rapid application development, and agile. Each model provides a framework to build reliable and scalable systems for specific project requirements. The development process aims toward ensuring functionality, usability, and efficiency, thus meeting the system objectives successfully.

4.2 Process Description

Different processes of each module are given below:

- Admin Login

The admin would have to enter the predefined email ID and password to access the system. On successful login, an admin can manage the institution - add or delete - as may be required. The institution changes instantly reflect only ensuring the user will get a real-time list of institutions for submission.

- User Registration and Data Submission

Participants submit their details through the system, providing necessary information on their participation in various initiatives. This system will maintain the information intelligently in reducing manual entry and will update the information in real time.

- Data Visualization

The collected data are processed and displayed in a systematic format by the system. The admin would be able to view the participation statistics and trends at

regular intervals which would be helpful in analyzing the progress in the generation of reports for better decision making.

4.3 Code Design

```
import 'package:flutter/material.dart';
import 'package:firebase_database/firebase_database.dart';
import 'line_chart.dart';
import 'dashboard_pages/ban_the_bag.dart';
import 'dashboard_pages/green_protocol.dart';
import 'dashboard_pages/sustainable_menstruation.dart';
import 'drawer.dart';

class DashboardPage extends StatefulWidget {
 const DashboardPage({super.key});
 @override
 _DashboardPageState createState() => _DashboardPageState();
}

class _DashboardPageState extends State<DashboardPage> {
 final DatabaseReference _banTheBagRef =
 FirebaseDatabase.instance.ref('ban_the_bag_form');
 final DatabaseReference _sustainableMenstruationRef =
 FirebaseDatabase.instance.ref('sustainable_menstruation_form');
 final DatabaseReference _institutionsRef =
 FirebaseDatabase.instance.ref('institutions');

 List<double> banTheBagCounts = List.filled(7, 0); // For the last 7 months
 List<double> sustainableMenstruationCounts =
 List.filled(7, 0); // For the last 7 months
 List<String> monthNames = []; // To hold month names for x-axis
 List<String> institutions = [];
 @override
```

```
void initState() {
  super.initState();
  initializeMonthNames();
  fetchData(); // Fetch data after initializing month names
  fetchInstitutions();
}
// Initialize month names for the last 7 months
void initializeMonthNames() {
  DateTime now = DateTime.now();
  monthNames = List.generate(7, (index) {
    DateTime monthDate = DateTime(now.year, now.month - index);
    return getMonthName(monthDate.month);
  }).reversed.toList();
}
// Convert month number to month name
String getMonthName(int month) {
  switch (month) {
    case 1:
      return 'Jan';
    case 2:
      return 'Feb';
    case 3:
      return 'Mar';
    case 4:
      return 'Apr';
    case 5:
      return 'May';
    case 6:
      return 'Jun';
    case 7:
      return 'Jul';
    case 8:
      return 'Aug';
    case 9:
```

```
      return 'Sep';
    case 10:
      return 'Oct';
    case 11:
      return 'Nov';
    case 12:
      return 'Dec';
    default:
      return '';
  }
}
// Fetch data for both Ban the Bag and Sustainable Menstruation
void fetchData() {
  resetCounts(); // Reset counts before starting any processing
  DateTime now = DateTime.now();
  _banTheBagRef.onValue.listen((DatabaseEvent event) {
    if (event.snapshot.exists) {
      Map<dynamic, dynamic> data =
      event.snapshot.value as Map<dynamic, dynamic>;
      for (var entry in data.entries) {
        if (entry.value['willing_to_participate'] == 'Yes') {
          var timestamp = entry.value['timestamp'];
          DateTime? date = _parseTimestamp(timestamp, now);
          if (date != null) {
            int monthIndex =
                (now.year - date.year) * 12 + now.month - date.month;
            if (monthIndex >= 0 && monthIndex < 7) {
              banTheBagCounts[6 - monthIndex]++;
            }
          }
        }
      }
      setState(() {});
    } else {
```

```
      print('No Ban The Bag data available.');
    }
  });
  _sustainableMenstruationRef.onValue.listen((DatabaseEvent event) {
    if (event.snapshot.exists) {
      Map<dynamic, dynamic> data =
      event.snapshot.value as Map<dynamic, dynamic>;
      for (var entry in data.entries) {
        if (entry.value['willing_to_participate'] == 'Yes') {
          var timestamp = entry.value['timestamp'];
          DateTime? date = _parseTimestamp(timestamp, now);
          if (date != null) {
            int monthIndex =
                (now.year - date.year) * 12 + now.month - date.month;
            if (monthIndex >= 0 && monthIndex < 7) {
              sustainableMenstruationCounts[6 - monthIndex]++;
            }
          }
        }
      }
      setState(() {}); // Update UI
    } else {
      print('No Sustainable Menstruation data available.');
    }
  });
}
// Fetch institution names from Firebase
void fetchInstitutions() {
  _institutionsRef.onValue.listen((DatabaseEvent event) {
    if (event.snapshot.exists) {
      Map<dynamic, dynamic> data =
      event.snapshot.value as Map<dynamic, dynamic>;
      institutions =
      List<String>.from(data.entries.map((entry) => entry.value ?? ''));
```

```
        setState(() {});
      } else {
        print('No institutions data available.');
      }
    });
  }
// Helper function to parse timestamp
  DateTime? _parseTimestamp(dynamic timestamp, DateTime fallback) {
    if (timestamp is int) {
      return DateTime.fromMillisecondsSinceEpoch(timestamp);
    } else if (timestamp is String) {
      return DateTime.tryParse(timestamp);
    }
    return null; // Return null for unsupported formats
  }
  // Reset counts before recalculating
  void resetCounts() {
    banTheBagCounts.fillRange(0, banTheBagCounts.length, 0);
    sustainableMenstruationCounts.fillRange(
        0, sustainableMenstruationCounts.length, 0);
  }
  @override
  Widget build(BuildContext context) {
    double screenWidth = MediaQuery.of(context).size.width;
    double screenHeight = MediaQuery.of(context).size.height;
    return Scaffold(
      drawer: AppDrawer(),
      appBar: AppBar(
        leading: Builder(
          builder: (BuildContext context) {
            return IconButton(
              icon: const Icon(Icons.menu),
              onPressed: () {
                Scaffold.of(context).openDrawer();
```

```
            },
          );
        },
      ),
      title: Center(
        child: Image.asset(
          'assets/logo.png',
          fit: BoxFit.contain,
          height: screenHeight * 0.05, // Adjusted height
        ),
      ),
    ),
    body: SingleChildScrollView(
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: <Widget>[
          SizedBox(height: screenHeight * 0.025),
          Padding(
            padding: EdgeInsets.symmetric(horizontal: screenWidth * 0.04),
            child: Text(
              'Challenge Dashboard',
              style: TextStyle(
                fontSize: screenWidth * 0.06, // Responsive font size
                fontWeight: FontWeight.bold,
                color: Colors.black,
              ),
            ),
          ),
          SizedBox(height: screenHeight * 0.02),
          _buildDashboardCard(
            context,
            "Total Number of Participants",
            '${banTheBagCounts.reduce((a, b) => a + b).toInt() +
sustainableMenstruationCounts.reduce((a, b) => a + b).toInt()}',
```

```
            ),
            _buildCategorySection(context),
            LineChartWidget(
              banTheBagCounts: banTheBagCounts,
              sustainableMenstruationCounts: sustainableMenstruationCounts,
              monthNames: monthNames,
            ),
            _buildInstitutionList(context),
            _buildClickableBoxes(context),
          ],
        ),
      ),
    );
  }
// Responsive Dashboard Card
  Widget _buildDashboardCard(BuildContext context, String title, String value) {
    double screenWidth = MediaQuery.of(context).size.width;
    double screenHeight = MediaQuery.of(context).size.height;
    return Padding(
      padding: EdgeInsets.symmetric(horizontal: screenWidth * 0.04),
      child: Container(
        width: screenWidth * 0.9, // 90% of screen width
        height: screenHeight * 0.2, // 20% of screen height
        decoration: BoxDecoration(
          color: const Color.fromARGB(255, 3, 98, 18),
          borderRadius: BorderRadius.circular(15.0),
        ),
        child: Padding(
          padding: EdgeInsets.all(screenWidth * 0.03),
          child: Column(
            crossAxisAlignment: CrossAxisAlignment.start,
            mainAxisAlignment: MainAxisAlignment.center,
            children: <Widget>[
              Text(
```

```
                title,
              style: TextStyle(
                color: Colors.white,
                fontSize: screenWidth * 0.04, // Scales dynamically
                fontWeight: FontWeight.bold,
              ),
            ),
            SizedBox(height: screenHeight * 0.005),
            Text(
              value,
              style: TextStyle(
                color: Colors.white,
                fontSize: screenWidth * 0.08, // Scales dynamically
                fontWeight: FontWeight.bold,
              ),
            ),
          ],
        ),
      ),
    ),
  );
 }
// Responsive Institution List
 Widget _buildInstitutionList(BuildContext context) {
  double screenWidth = MediaQuery.of(context).size.width;
  double screenHeight = MediaQuery.of(context).size.height;
  return Padding(
   padding: EdgeInsets.symmetric(horizontal: screenWidth * 0.04),
   child: Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: [
     SizedBox(height: screenHeight * 0.025),
     Text(
      'Institutions',
```

```
  style: TextStyle(
    fontSize: screenWidth * 0.05,
    fontWeight: FontWeight.bold,
    color: Colors.black,
  ),
),
SizedBox(height: screenHeight * 0.015),
Container(
  height: screenHeight * 0.3, // 40% of screen height
  decoration: BoxDecoration(
    color: Colors.grey.withOpacity(0.1),
    borderRadius: BorderRadius.circular(12.0),
    boxShadow: [
      BoxShadow(
        color: Colors.grey.withOpacity(0.2),
        spreadRadius: 3,
        blurRadius: 6,
        offset: Offset(0, 2),
      ),
    ],
  ),
  child: institutions.isEmpty
      ? const Center(child: Text("No institutions available"))
      : ListView.builder(
    physics: const BouncingScrollPhysics(),
    itemCount: institutions.length,
    itemBuilder: (context, index) {
      return Card(
        elevation: 3,
        margin: EdgeInsets.symmetric(
            vertical: screenHeight * 0.01,
            horizontal: screenWidth * 0.025),
        shape: RoundedRectangleBorder(
          borderRadius: BorderRadius.circular(8.0),
```

```
),
child: ListTile(
  contentPadding: EdgeInsets.symmetric(
    vertical: screenHeight * 0.01,
    horizontal: screenWidth * 0.04,
  ),
  title: Row(
    children: [
      Text(
        '${index + 1}. ',
        style: TextStyle(
          fontSize: screenWidth * 0.045,
          fontWeight: FontWeight.w500,
        ),
      ),
      Expanded(
        child: Text(
          institutions[index],
          style: TextStyle(
            fontSize: screenWidth * 0.045,
            fontWeight: FontWeight.w500,
          ),
          maxLines: 1,
          overflow: TextOverflow.ellipsis,
        ),
      ),
      Icon(
        Icons.school,
        color: Color.fromARGB(255, 3, 98, 18),
        size: screenWidth * 0.05,
      ),
    ],
  ),
),
```

```
                );
              },
            ),
          ),
        ],
      ),
    );
  }
// Responsive Category Section
  Widget _buildCategorySection(BuildContext context) {
    double screenWidth = MediaQuery.of(context).size.width;
    double screenHeight = MediaQuery.of(context).size.height;
    return Column(
      children: <Widget>[
        SizedBox(height: screenHeight * 0.025),
        Padding(
          padding: EdgeInsets.symmetric(horizontal: screenWidth * 0.04),
          child: Text(
            'Current Impact',
            style: TextStyle(
              fontSize: screenWidth * 0.05,
              fontWeight: FontWeight.bold,
              color: Colors.black,
            ),
          ),
        ),
        SizedBox(height: screenHeight * 0.015),
      ],
    );
  }
  Widget _buildClickableBoxes(BuildContext context) {
    double screenWidth = MediaQuery.of(context).size.width;
    double screenHeight = MediaQuery.of(context).size.height;
    return Padding(
```

```
padding: EdgeInsets.symmetric(
   horizontal: screenWidth * 0.04, vertical: screenHeight * 0.025),
child: Column(
 crossAxisAlignment: CrossAxisAlignment.start,
 children: [
  SizedBox(height: screenHeight * 0.025), // Responsive spacing
  Text(
   'Explore Challenges',
   style: TextStyle(
      fontSize: screenWidth * 0.05, // Responsive font size
      fontWeight: FontWeight.bold,
      color: Colors.black),
  ),
  SizedBox(height: screenHeight * 0.015),
  Center(
   child: Row(
    mainAxisAlignment: MainAxisAlignment.center,
    children: <Widget>[
     Flexible(
      child: _buildClickableBox(
         context,
         'Green Protocol',
         'assets/green.png',
         const GreenProtocol(message: 'Hello1')),
     ),
     SizedBox(width: screenWidth * 0.02), // Reduced spacing
     Flexible(
      child: _buildClickableBox(
         context,
         'Sustainable Menstruation',
         'assets/sustainable_mensuration.png',
         const SustainableMenstruation(message: 'Hello2')),
     ),
     SizedBox(width: screenWidth * 0.02),
```

```
                Flexible(
                  child: _buildClickableBox(
                      context,
                      'Ban the Bag',
                      'assets/ban_the_bag.png',
                      const BanTheBag(message: 'Hello3')),
                ),
              ],
            ),
          ),
          SizedBox(height: screenHeight * 0.025),
        ],
      ),
    );
  }
// Single clickable box
  Widget _buildClickableBox(
      BuildContext context, String title, String imagePath, Widget page) {
    double screenWidth = MediaQuery.of(context).size.width;
    double screenHeight = MediaQuery.of(context).size.height;
    return GestureDetector(
      onTap: () {
        Navigator.push(
          context,
          MaterialPageRoute(builder: (context) => page),
        );
      },
      child: Container(
        width: screenWidth * 0.3, // Responsive width
        height: screenHeight * 0.18, // Responsive height
        decoration: BoxDecoration(
            color: Colors.white,
            borderRadius: BorderRadius.circular(screenWidth * 0.04), // Responsive
border radius
```

```
            boxShadow: [
              BoxShadow(
                color: Colors.grey.withOpacity(0.4),
                spreadRadius: 2,
                blurRadius: 5,
                offset: Offset(0, 2),
              ),
            ]),
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              Image.asset(imagePath, width: screenWidth * 0.12),
              Text(title,
                  textAlign: TextAlign.center,
                  style: TextStyle(
                      fontSize: screenWidth * 0.03, // Responsive text size
                      fontWeight: FontWeight.bold,
                      color: Colors.black)),
            ],
          ),
        ),
      );
    }
  }
```

5. SYSTEM TESTING AND IMPLEMENTATION

5.1 Introduction

Software testing is a crucial process in the software development life cycle that involves evaluating the quality and functionality of software applications. The main goal of software testing is to identify defects or errors in the application before it is released to the end-users. The software testing process typically involves several steps, including test planning, test design, test execution, and test reporting.

5.2 System Implementation

Implementation is the action that must follow any preliminary thinking for something to happen. Software/hardware implementations should always be designed with the end user in mind and the implementation process usually benefits from user involvement and support from managers and other top executives in the company. If users participate in the design and implementation of the system, ideally it will serve their business objectives more accurately and reflect their priorities and the ways in which they prefer to work.

5.3 Debugging

Debugging is the process of identifying and resolving issues or errors in software applications. It is a critical step in the software development life cycle and is aimed at improving the quality and functionality of the application. During the development process, errors or bugs can occur in the code that can cause the application to behave in unexpected ways or not work at all. Debugging involves identifying and diagnosing these errors, tracing their root cause, and then making the necessary changes to fix them.

Different types of debugging methods used in this system are:

- Unit Testing: The application was divided into smaller components and tested individually. Each code was executed separately to ensure accuracy.

- Integration Testing: Each small component was integrated or combined into a module to ensure that each module works properly when put together. This was done to check connectivity between modules.
- System Testing: The system was tested by combining every module. This was to ensure that each process had a particular order. This was to ensure that the system does not crash while using.

5.4 System Security

Password encryption is used to protect each user's details. Administrators have separate user profiles that allow viewing of collective data.

5.5 Scope for Future Enhancements

For future enhancements, the app could use AI-powered analytics to provide predictive insights on sustainability trends and allow offline data entry for users to submit data without an internet connection. Integration with government ldatabases can enhance data verification, and automated notifications can improve engagement by reminding users to participate in initiatives. Expanding the app to track additional sustainability metrics can further support environmental and public health efforts.

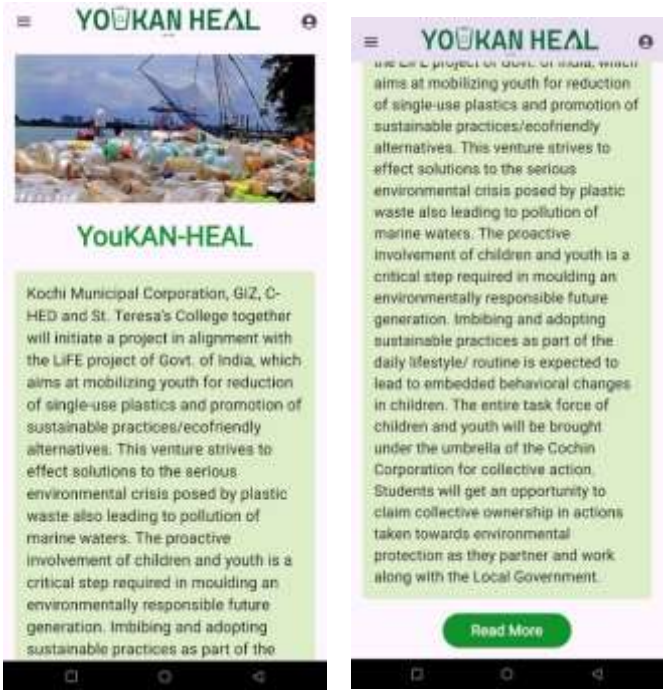
6. CONCLUSION

The YouKan Heal Dashboard App plays an important role in promoting sustainable practices by providing a digital platform for institutions and individuals to participate in various environmental challenges. The system enables real-time data tracking, interactive visualization, and seamless participation in programs like Ban the Bag, Sustainable Menstruation, and Green Protocol.

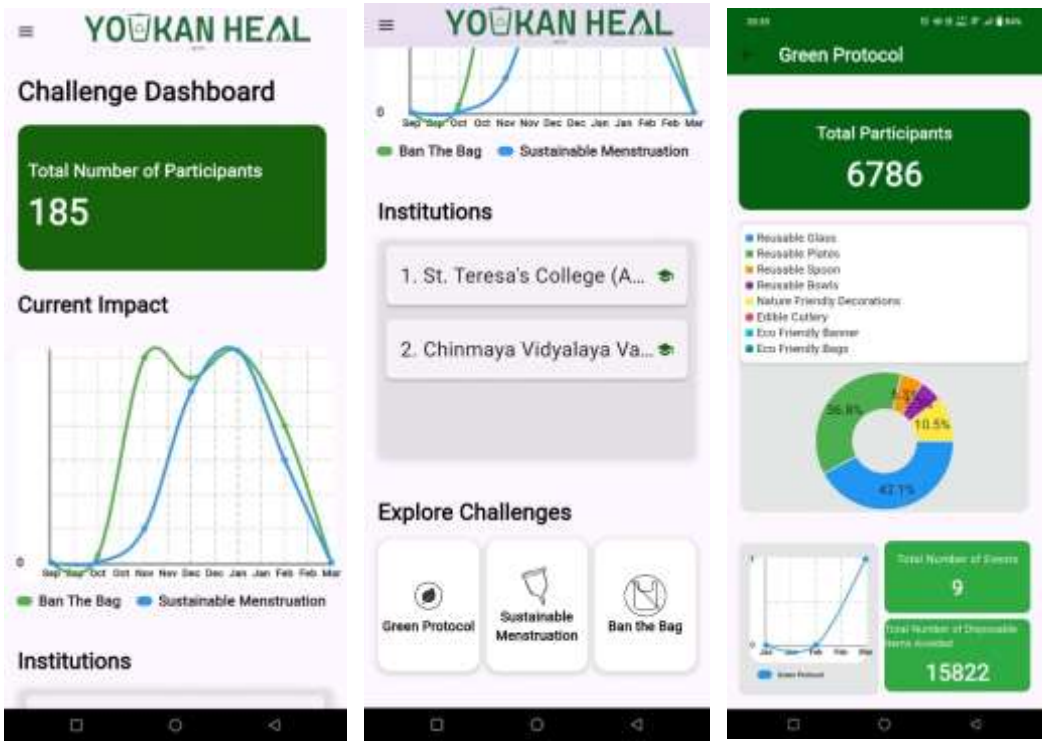
Through its usability and management utilities, the app makes data collection and management easier, fostering stakeholder involvement with sustainability. All these product features translate to better technology for better traction in changing the planetary destiny, therefore, contributing to a resourceful eco-community striving towards sustainability.

7. APPENDIX

→ Home Page



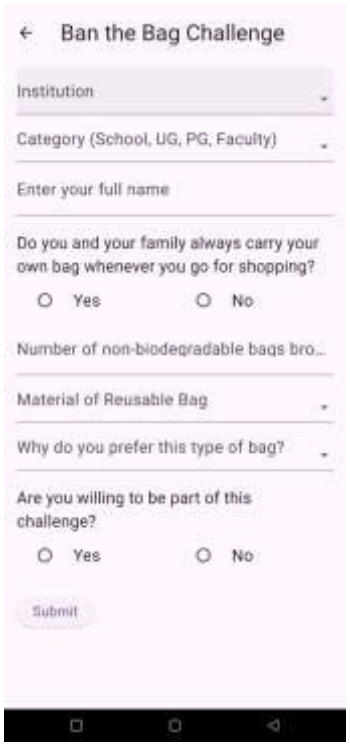
→ Dashboard

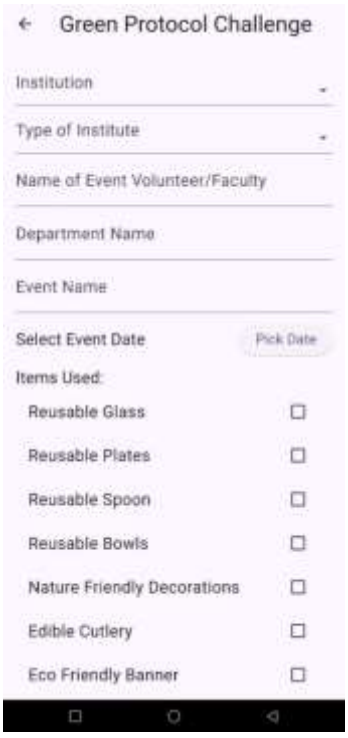


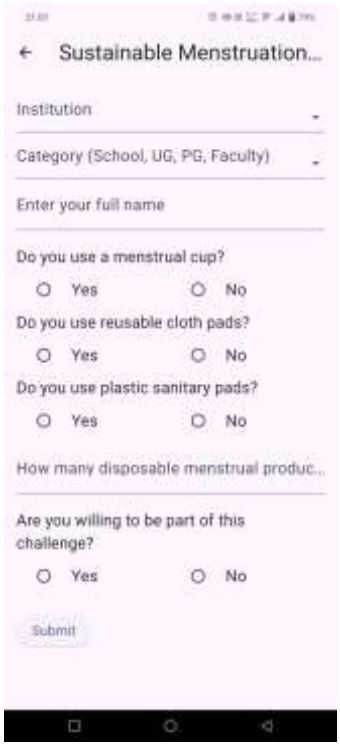
→ User Login


Login
Email
Password
Login
Create New Account


Choose a Challenge
Ban the Bag Challenge
Green Protocol Challenge
Sustainable Menstruation Challenge


Ban the Bag Challenge
Institution
Category (School, UG, PG, Faculty)
Enter your full name
Do you and your family always carry your own bag whenever you go for shopping?
Yes
No
Number of non-biodegradable bags bro...
Material of Reusable Bag
Why do you prefer this type of bag?
Are you willing to be part of this challenge?
Yes
No
Submit


Green Protocol Challenge
Institution
Type of Institute
Name of Event Volunteer/Faculty
Department Name
Event Name
Select Event Date
Pick Date
Items Used:
Reusable Glass
Reusable Plates
Reusable Spoon
Reusable Bowls
Nature Friendly Decorations
Edible Cutlery
Eco Friendly Banner

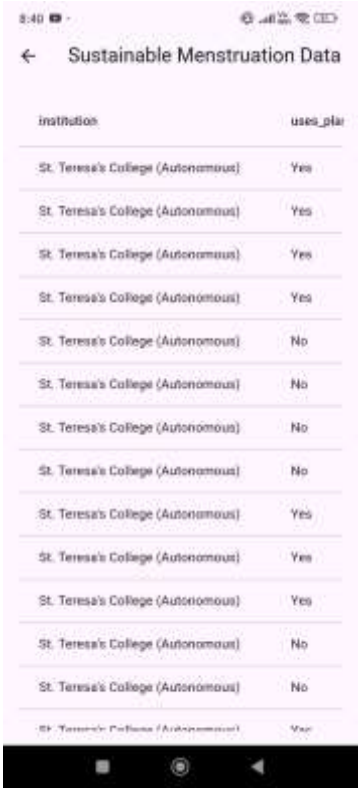

Sustainable Menstruation...
Institution
Category (School, UG, PG, Faculty)
Enter your full name
Do you use a menstrual cup?
Yes
No
Do you use reusable cloth pads?
Yes
No
Do you use plastic sanitary pads?
Yes
No
How many disposable menstrual produc...
Are you willing to be part of this challenge?
Yes
No
Submit

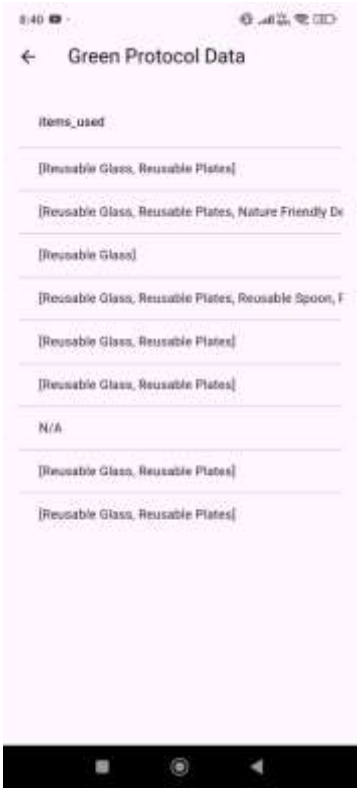
→ Admin Login

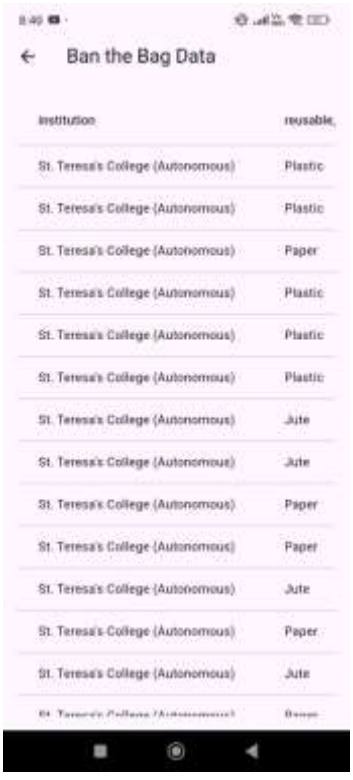

YOUKAN HEAL
Admin Dashboard
Add Institutions
Show Data


Manage Institutions
Institution Name
Add Institution
Institutions
St. Teresa's College (Autonomous)
Chinmaya Vidyalaya Vaduthala


Data Overview
Ban the Bag Data
Sustainable Menstruation Data
Green Protocol Data


Sustainable Menstruation Data
institution
St. Teresa's College (Autonomous)
Yes
No


Green Protocol Data
items_used
[Reusable Glass, Reusable Plates]
[Reusable Glass]
N/A


Ban the Bag Data
institution
St. Teresa's College (Autonomous)
Plastic
Paper
Jute

8. BIBLIOGRAPHY

- YouKan Heal Official Website: https://mykochi.lsgkerala.gov.in/youkanheal
- Flutter Documentation – https://flutter.dev/docs
- Firebase Documentation – https://firebase.google.com/docs