

**GROUP 6**  
**WATERWISE**  
**BSINFOTECH 302B**

**SUBMITTED TO:**  
**Vince Lorence Dela Rea**

**SUBMITTED BY:**  
**Jeiya Cazandra de Joya**  
**Kayla Denise Dulce**  
**Mark Vincent Jimenez**  
**Joshua Carillo**

# WaterWise

**Description:** A simple app that helps you track how much water you use every day. It gives you easy tips on saving water and allows you to create a limit for yourself to watch over water usage.

## Function:

### Daily Water Tracking:

- Users manually input their water usage.
- The app estimates water consumption based on water usage entered.

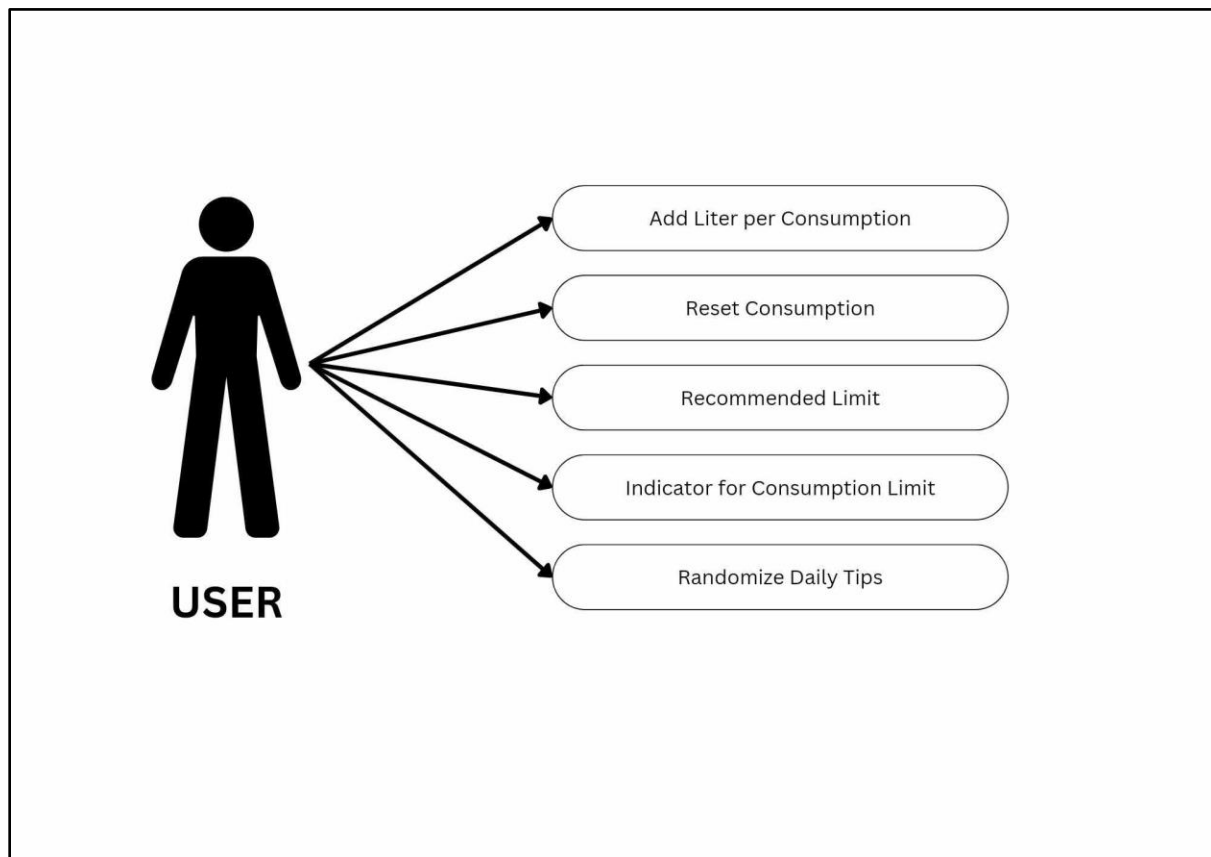
### Daily Tips:

- The app provides one simple tip each day to help users save water, such as "Turn off the tap while brushing your teeth."
- Tips are easy to follow and require minimal effort.

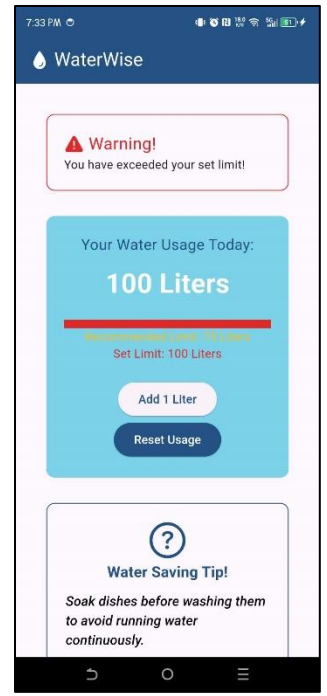
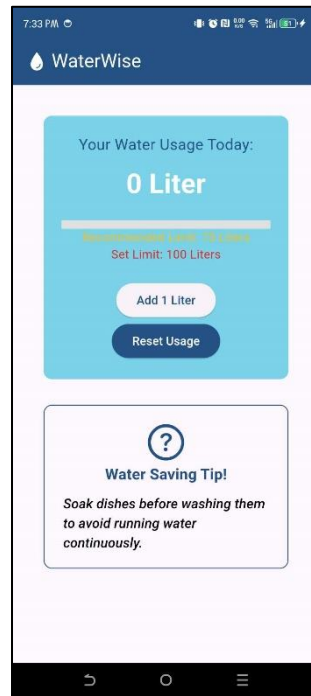
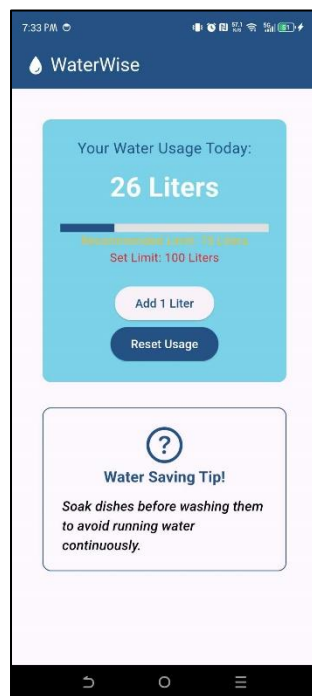
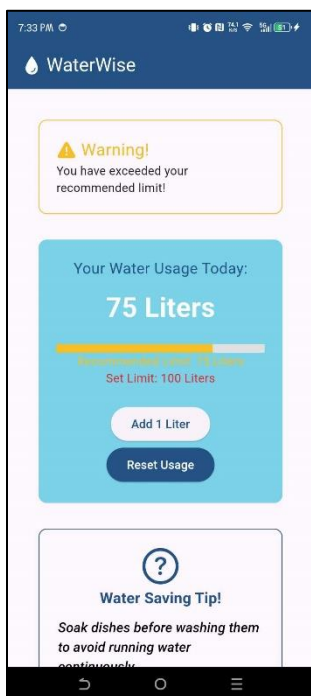
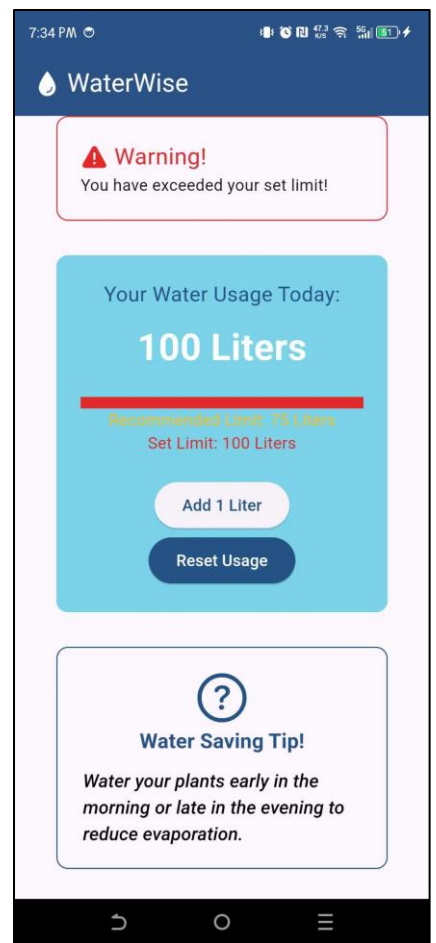
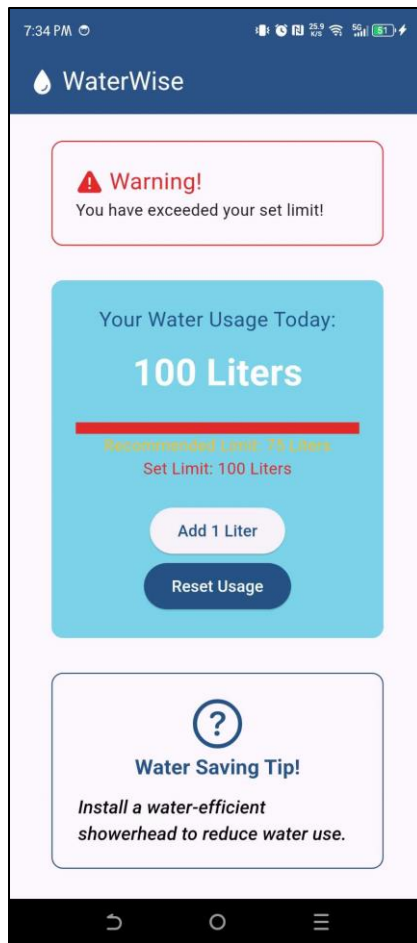
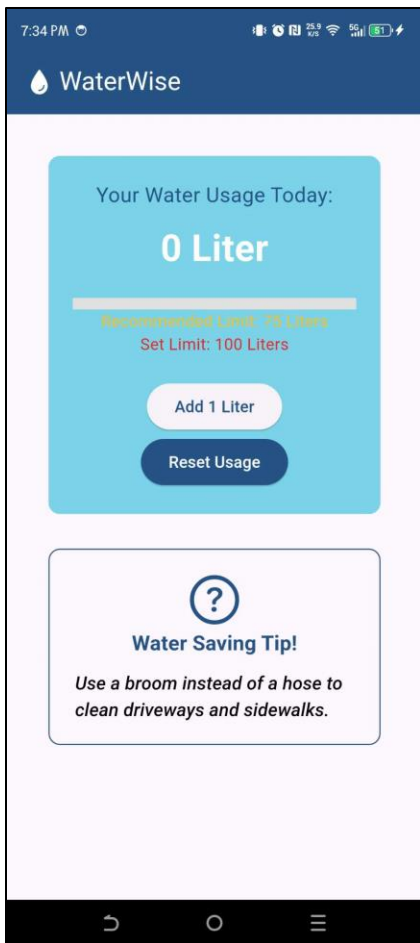
### Simple Progress Tracking:

- A progress bar shows how close they are to their water-saving goal.

## Sample use-case of the App:



## Screenshot of the App:



### Source Code:

```
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:shared_preferences/shared_preferences.dart';
import 'dart:math';

void main() {
  runApp(WaterWiseApp());
}

class WaterWiseApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'WaterWise',
      theme: ThemeData(
        primarySwatch: Colors.blue,
      ),
      home: WaterUsageScreen(),
    );
  }
}

class WaterUsageScreen extends StatefulWidget {
  @override
  _WaterUsageScreenState createState() => _WaterUsageScreenState();
}

class _WaterUsageScreenState extends State<WaterUsageScreen> with WidgetsBindingObserver {
  int _waterUsage = 0;
  bool limVisible = false;
  int _hardLimit = 100;
  int _recLimit = 75;
  String limRem = "";
  int limColor = 0xFFf7bf25;
  List<String> dailyTip = [
    "Turn off the tap while brushing your teeth.",
    "Use a bucket to wash your car instead of a hose.",
    "Fix any leaks in your faucets or pipes to avoid wasting water.",
    "Only run the dishwasher when it's full to save water.",
    "Collect rainwater to water your plants instead of using tap water.",
    "Use a broom instead of a hose to clean driveways and sidewalks.",
    "Install a water-efficient showerhead to reduce water use.",
    "Water your plants early in the morning or late in the evening to reduce evaporation.",
    "Soak dishes before washing them to avoid running water continuously."];
  int _tip = Random().nextInt(9); // Randomizes the tip the user may get every time they open the app.

  @override
```

```

void initState() {
    super.initState();
    _loadWaterUsage();
}

// Load saved water usage from SharedPreferences.
void _loadWaterUsage() async {
    SharedPreferences prefs = await SharedPreferences.getInstance();
    setState(() {
        _waterUsage = (prefs.getInt('waterUsage') ?? 0);
    });
}

// Save water usage to SharedPreferences.
void _saveWaterUsage(int usage) async {
    SharedPreferences prefs = await SharedPreferences.getInstance();
    setState(() {
        _waterUsage = usage;
        prefs.setInt('waterUsage', usage);
    });
}

// Adds 1 Liter to the WaterUsage.
void _incrementUsage() {
    _waterUsage += 1;
    _saveWaterUsage(_waterUsage);
    limitReminder();
}

// Resets the WaterUsage back to 0 Liters.
void _resetUsage() {
    _waterUsage = 0;
    _saveWaterUsage(_waterUsage);
    limitReminder();
}

// Contains details regarding which color and text to display when exceeding limits.
void limitReminder(){
    if(_waterUsage >= _recLimit){
        if(_waterUsage >= _hardLimit){
            limVisible = true;
            limColor = 0xFFe32929;
            limRem = 'You have exceeded your set limit!';
        } else {
            limVisible = true;
            limColor = 0xFFf7bf25;
            limRem = 'You have exceeded your recommended limit!';
        }
    } else {

```

```

    limVisible = false;
  };
}

```

```
@override
```

```

Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      backgroundColor: Color(0xFF265385),
      title: Row(
        children: [
          Icon(
            CupertinoIcons.drop_fill,
            color: Colors.white
          ),
          Text(' WaterWise',
            style: TextStyle(
              color: Colors.white
            )
          )
        ],
      ),
    ),
    body: SingleChildScrollView(
      padding: const EdgeInsets.all(16.0),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          Offstage(
            offstage: !limVisible, // Keeps the alert/warning box to be hidden until the WaterUsage exceeds the
set limits.
            child: Container(
              margin: EdgeInsets.only(left:20.0,right:20.0,top:20.0,bottom:10.00),
              padding: EdgeInsets.all(20.0),
              decoration: BoxDecoration(
                borderRadius: BorderRadius.circular(10),
                border:Border.all(
                  color: Color(limColor)
                )
              ),
            ),
            child: Row(
              children: [
                Expanded(
                  child: Column(crossAxisAlignment: CrossAxisAlignment.start,
                    children: [
                      Row(
                        children: [
                          Icon(CupertinoIcons.exclamationmark_triangle_fill,

```

```

        color: Color(limColor)
      ),
      Text(' Warning!',
        style: TextStyle(
          fontSize: 20,
          color: Color(limColor)
        )
      ),
    ],
  ),
  Text(limRem)
]
)
)
],
)
),
),
Container( // Main container that has the current WaterUsage, information about the set limits, the
progress bar, and the buttons.
  margin: EdgeInsets.all(20.0),
  padding: EdgeInsets.all(20.0),
  decoration: BoxDecoration(
    color: Color(0xFF7BD3EA),
    borderRadius: BorderRadius.circular(10),
    border: Border.all(
      color: Color(0xFF7BD3EA)
    )
  ),
  child: Column(
    children: [
      Row(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          Text(
            'Your Water Usage Today:',
            style: TextStyle(
              fontSize: 18,
              color: Color(0xFF265385)
            )
          )
        ]
      ),
      SizedBox(height: 10),
      Row(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          Text(
            '_waterUsage > 1 ? '$_waterUsage Liters' : '$_waterUsage Liter',

```

```

        style: TextStyle(
          fontSize: 32,
          fontWeight: FontWeight.bold,
          color: Colors.white
        )
      ),
    ],
  ),
  SizedBox(height: 20),
  Container(
    child: LinearProgressIndicator(
      value: _waterUsage / _hardLimit,
      minHeight: 10,
      backgroundColor: Colors.grey[300],
      valueColor: AlwaysStoppedAnimation<Color>(
        Color(
          _waterUsage >= _hardLimit ? 0xFFe32929 : _waterUsage >= _recLimit? 0xFFf7bf25 :
0xFF265385
        )
      ),
    ),
  ),
  Row(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      Column(
        children: [
          Text('Recommended Limit: $_recLimit Liters',
            style: TextStyle(
              color: Color(0xFFf7bf25)
            )
          ),
          Text('Set Limit: $_hardLimit Liters',
            style: TextStyle(
              color: Color(0xFFe32929)
            )
          )
        ]
      )
    ],
  ),
  SizedBox(height: 20),
  Row(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      ElevatedButton(
        onPressed: _incrementUsage,
        child: Text('Add 1 Liter', style: TextStyle(color: Color(0xFF265385))),
      ),
    ],
  ),

```



```

    ],
  ),
  Row(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      ElevatedButton(
        style: ElevatedButton.styleFrom(
          backgroundColor: Color(0xFF265385)
        ),
        onPressed: _resetUsage,
        child: Text('Reset Usage',
          style: TextStyle(color: Colors.white)
        )
      )
    ],
  ),
]
)
),
Container(
  margin: EdgeInsets.only(left:20.0,right:20.0,top:10.0,bottom:10.00),
  padding: EdgeInsets.all(10.0),
  decoration: BoxDecoration(
    borderRadius: BorderRadius.circular(10),
    border: Border.all(
      color: Color(0xFF265385)
    )
  ),
),
child: Row(
  children: [
    Expanded( // Tapping on the Water Saving Tip box allows you to randomize a new tip.
      child: TextButton(onPressed:() {
        setState() {
          int newtip = _tip;
          while(newtip == _tip){
            newtip = Random().nextInt(9);
          }
          _tip = newtip;
        });
      },
      child: Column(
        children: [
          Icon(CupertinoIcons.question_circle,
            color: Color(0xFF265385),
            size: 50
          ),
          Text('Water Saving Tip!',
            style: TextStyle(
              fontSize: 18,

```

[illegible]