# DECLARATION

We,**SINCHANA A B, SINCHANA A S, SWATHI M N and TANUSHREE N P** students of Eighth semester B.E**, GOVERNMENT ENGINEERING COLLEGE, HASSAN** bearing USN **4GH21CS044 ,4GH21CS045**, **4GH21CS054** and **4GH21CS055** respectively, hereby declare that the Project entitled “**KRISHI MITRA PORTAL”** has been carried out by our team under the supervision of our Guide, **Dr. Vani V G** B.E., MTech, Department of CS&E, GEC Hassan, have submitted in partial fulfilment of the requirements for the award of the Degree of B.E in CS&E by the Visvesvaraya Technological University, Belagavi during the academic year 2024- 2025. This report has not been submitted to any other Organization/University for the award of degree or certificate .

**Date: Project Associates**

**Place: Hassan** SINCHANA A B

SINCHANA A S

SWATHI M N

TANUSHREE N P

**ACKNOWLEDGEMENT**

We consider it a privilege to whole-heartedly express our gratitude and respect to each and every one who guided and helped us in the successful completion of this Project Report.

We very thankful to the Principal **Dr. Girish D P,** for being kind enough to provide me an opportunity to work on a project in this institution.

We also thankful to **Dr. Vani V G**, HOD, Department of Computer Science, for her co-operation and encouragement at all moments of our approach.

We would greatly mention the enthusiastic influence provided by **Dr.Vani V G,** Assistant Professors, as Project Guide for his ideas and co-operation showed on us during our venture and making this Project as a great success.

We also thankful to **Prof. Kiran M P**, Project Co-Ordinator, Department of Computer Science and Engineering for the Co-Operations and encouragement at all the moments of our approach.

We would also like to thank our parents and well-wishers as well as our dear classmates for their guidance and their kind co-operation.

Finally, it is our pleasure and happiness to the friendly co-operation showed by all the staff members of Computer Science Department, GECH.

**Project associates:**

Sinchana A B (4GH21CS044)

Sinchana A S (4GH21CS045)

Swathi M N (4GH21CS054)

Tanushree N P (4GH21CS055)

**i**

**ABSTRACT**

The Agricultural Portal is an innovative platform designed to improve crop production by providing farmers with easy access to agricultural information, resources, and tools. The portal offers a wide range of features including weather forecasts, pest and disease management tips, soil health assessment, crop planning tools, and market prices. This technical paper outlines the development and implementation of the Agricultural Portal, highlighting its features and functionalities. The paper also explores the benefits of the portal for farmers, including increased productivity, improved decision-making, and enhanced profitability. The portal is built on a robust technology platform that is scalable and adaptable to the needs of farmers of different sizes and geographies. It is designed to be user-friendly and accessible on multiple devices, including mobile phones and tablets. The Agricultural Portal represents a significant step forward in the use of technology in agriculture. By providing farmers with easy access to information and resources, it has the potential to transform the way they farm and improve crop production across the globe. Keywords-Agricultural portal, crop production, farmers, user friendly.

**ii**

**TABLE OF CONTENTS**

**Acknowledgment i**

**Abstract ii**

**Table of Content iii**

**List of Figures vi**

1. **INTRODUCTION 01**

## 1.1 Project Overview 02

1.2 Scope 02

1.3 Challenges 02

1.4 Objectives 02

**2. LITERATURE SURVEY 03**

2.1 Phishing Detection: Analysis of Visual Similarity Based Approaches 03

2.2 Phishing Website Detection based on Supervised Machine Learning with Wrapper Features Selection 04

2.3 Detecting Phishing Website Using Machine Learning 05

2.4 Phishing website detection using machine learning and deep learning techniques 05

**3. REQUIREMENTS SPECIFICATION 07**

3.1 Functional Requirements 07

3.2 Non-Functional Requirements 08

3.3 Installation Requirements 08

**iii**

3.3.1 Hardware Requirements 08

3.3.2 Software Requirements 08

**4. SYSTEM DESIGN 09**

4.1 Anaconda Navigator 09

4.1.1 Jupyter Notebook 09

4.2 Django 12

**5. ARCHITECTURE 13**

5.1 System Architecture 13

5.2 Flowchart Of The System 14

5.3 Data Collection 16

5.4 Data Pre-Processing 16

5.5 Exploratory Data Analysis 16

5.6 Feature Extraction 17

5.7 Training And Testing The Model 17

5.8 Model Evaluation 19

**6. IMPLEMENTATION 20**

6.1 Pseudo Code 20

6.2 Data Collection 22

6.3 Feature Extraction On The Datasets 23

6.4 Data Analysis & Visualization 27

6.5 Data Pre-Processing 29

6.6 Phishing Detection Model 31

6.7 General Working Of The System 32

**iv**

6.7.1 The Home Page 32

6.7.2 The About Page 32

6.7.3 The Use Case Page 32

6.7.4 Web Application Source Code 32

6.8 Api (application programming interface) 35

**7. RESULTS 38**

7.1 URL Entry In PhishDetect 38

7.2 Prediction 39

**8. CONCLUSION AND FUTURE SCOPE 41**

8.1 Conclusion 41 8.2 Future Scope 41

**REFERENCES 43**

**v**

# LIST OF FIGURES

# 5.1: Architectural Design Of The Proposed System 13

# 5.2.1: Flowchart Of The proposed System 14

# 5.2.2: Flowchart of the web interface 15

# 5.7.1: High loss in the left model; Low loss in the right model. 17

# 5.7.2: Slicing a single data set into a training set and test set. 18

# 6.2.1: Dataset of Phishing URLs 22

# 6.2.2: Dataset of Legitimate URLs 23

# 6.3.1: Code for Address bar based feature extraction 24

# 6.3.2: Code for Domain-based features extraction 25

# 6.3.3: Code for Html & java-script based features extraction 26

# 6.3.4: Code computation for all the Feature Extraction used dataset 27

# 6.4.1: Distribution plot of dataset base on the features selected 28

# 6.4.2: Correlation heat map of the dataset 29

# 6.5.1: Summary of the dataset 30

# 6.5.2: Number of missing values in the dataset 30

# 6.6: Accuracy performance of models 31

# 6.7.1: The Home page 33

# 6.7.2: The About page 33

# 6.7.3: The Use-case page 34

# 6.7.4: Code for the web application 34

# 6.8.1: Python code to send a request on JSON 35

# 6.8.2: Code for API URL 36

# 6.8.3: Executing web-app on Django local server 36

# 6.8.4: JavaScript code for linking API 37

# 7.1: Entering URL to PhishDetect 38

# 7.2 (a): Entered URL is Predicted 39

# 7.2 (b): Result of Predicted Output 40

# vi