

A mini project report submitted
on

E-MARKET FOR FARMERS

For the partial fulfillment of the requirements for award of the degree
of

**BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE & ENGINEERING**

Submitted by:

E.V.V.MOHINI

319129510015

A.VINEELA

319129510001

K.MOUNIKA

319129510024

Y.TIRUPATHI RAO

319129510051

V.VARA LAKSHMI

319129510048

Under the Esteemed Guidance of

Mr.K.V.KIRAN M.Tech(Ph.D)

Associate Professor ,Dept.of CSE, WISTM.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

WELFARE INSITUTE OF SCIENCE TECHNOLOGY AND MANAGEMENT

Approved by AICTE, New Delhi & affiliated to Andhra University Pinagadi (V),
Pendurthi (M), Visakhapatnam-531173.

WELFARE INSITUTE OF SCIENCE TECHNOLOGY AND MANAGEMENT

Approved by AICTE, New Delhi & affiliated to Andhra University

Pinagadi (V), pendhurthi(M), Visakhapatnam-531173



CERTIFICATE

This is to certify that the project report entitled “E-MARKET FOR FARMERS” being
submitted by

E.V.V.MOHINI

319129510015

A.VINEELA

319129510001

K.MOUNIKA

319129510024

Y.TIRUPATHI RAO

319129510051

V.VARA LAKSHMI

319129510048

For partial fulfilments for the award of the degree Bachelor of Technology in Computer science and Engineering is a bonafide project work carried out by him .This work is not submitted to any University for the award of any degree.

INTERNAL GUIDE

Mr.K.V.KIRAN M.Tech(Ph.D)
ASSOC.PROFESSOR,
DEPARTMENT OF CSE,
WISTM ENGG.COLLEGE

HEAD OF THE DEPARTMENT

Mrs. K. V.LAKSHMI, M .Tech
ASSISTANT PROFESSOR
DEPARTMENT OF CSE,
WISTM ENGG.COLLEGE

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people who made it possible and whose constant guidance and encouragement crown all the effort with success. I like to thank **prof. M. James Stephen, Principal**, of Wellfare institute of science, Technology and Management, for this kind cooperation that gives me boundless pleasure to avail this opportunity to express my deep sense of gratitude and whole hearted thanks to Head of Department **Mrs. K.V.Lakshmi, Assistant Professor**, Department of computer science and Engineering and my project guide Mr. **K.V.Kiran , Assoc. Professor** ,Department of computer science and Engineering, and all the faculty of computer science department for their valuable suggestions and co-operation during the development of this project. I am really grateful to my parents and all my friends for their constructive suggestions to do this project in effective manner.

DECLARATION

I here declare that the project entitled “**E-MARKET FOR FARMERS**” has been done under the esteemed guidance of **Mr. K .V .Kiran ,Assoc.professor**, Department of Computer Science and Engineering and is dissertation of my own work except where specifically ask to the contrary and is submitted to the department of Computer Science and Engineering, Welfare Institute of science Technology and Management for the partial fulfilment of the requirement for the award of B.Tech degree.

E.V.V.MOHINI(319129510015)

A.VINEELA(319129510001)

K.MOUNIKA(319129510024)

V.VARALAKSHMI(319129510048)

Y.TIRUPATHI RAO(319129510051)

E-MARKET FOR FARMERS

ABSTRACT

Around many years, farmers in the India have had a little freedom in choosing markets and buyers for their product. All states in the country, mandate that marketing and selling of the farm product must be routed through state owned man is, retail markets. In these days small scale farmers are not showing interest over farming. As if there are not getting their production rates. If farmers sell their products also they are not getting reasonable profits. So farmer are leaving farming fields. To make farmers easy we are going to provide an online portal for farmers to sell their products from their towns and villages etc. This portal creates a communication between the farmer and the small scale vendors .So the main aim of this portal is to create an interest regarding farming. In this farmers can upload their products like vegetables,fruits,cereals,spices, cattle etc. The small scale vendors can register and login into this page, and view products availability. The farmers can sell products to their reasonable costs, and if there are interest will give their cost if they need. Now farmer will select the person according to his willing price.So that both farmers and small scale farmer will get profits.

CONTENTS

TITLE	PAGE NO
1. INTRODUCTION	1
2. LITERATURE SURVEY	2
3. SYSTEM ANALYSIS	
3.1 Existing system	3
3.2 Proposed system	4
3.3 System requirements specifications	5-7
3.3.1 Functional Requirements	
3.3.2 Non-Functional Requirements	
4. SYSTEM REQUIREMENTS	7-20
4.1 Hardware Specification	
4.2 Software Specification	
4.3 Software Descriptions	
5. MODULE DESCRIPTION	21-22
6. SYSTEM DESIGN	23-34
6.1 System Architecture	
6.2 Introduction to UML	
6.3 Implementation	
6.4 UML Diagrams	
6.4.1 Use case Diagram	
6.4.2 Class Diagram	
6.4.3 Sequential Diagram	
6.4.4 component Diagram	
6.4.5 Deployment Diagram.	
7. CODING	35-63
8. SCREENSHOTS	64-70
9. CONCLUSION	71

1.INTRODUCTION

1.1 SCOPE AND PURPOSE:

E-MARKET FOR FARMERS is the web application that will help the farmers to perform the marketing leading to achieve success and increase in their standard of living. The Marketing facility would allow the farmers to have a view of the bills created and the related information in their accounts. An Authorized-agent would serve as a way for the farmers to sell their products in the market. Website will also provide commodity-wise report to the farmer in interactive way. All states in the country, mandate that marketing and selling of the farm product must be routed through state owned man is, retail markets. In these days small scale farmers are not showing interest over farming. As if there are not getting their production rates. If farmers sell their products also they are not getting reasonable profits. So farmer are leaving farming fields. To make farmers easy we are going to provide an online portal for farmers to sell their products from their towns and villages etc. This portal creates a communication between the farmer and the small scale vendors.

1.2 OVERALL SYSTEM DESIGN OBJECTIVES:

The main objective of this project is building a website which will help farmers from Indian villages to sell their products to different city markets. It is a computerized approach for better and clear marketing. The main objective of this project is that there is a direct communication is done in between the User and the farmer.

2. LETERATURE SURVEY

The farmers are facing many problems in their farming . farmers many are illiterate to know all the facilities provided by government and any natural disasters occurs then loss of land and their productions. If any farmer having large scale lands are benefited but small scale farmers not getting good profit and non-productivity rate for their products .So , they are not getting any interest regarding on farming. And move to towns for their survival.

3. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM:

Most of the food crops are carried into the market and sell in a particular day where intermediate persons are allowed for auction. But during these days the intermediate persons are forming a committee and held meeting them self before auction starts and sets the maximum allowed price for all type of crops based on current market price and their profit. After purchasing at lower price from farmers ,they intentionally create a demand in the market in next few days by storing goods in their storage facility and sell same at higher price to the end-users . This is the actual main hidden reason why farmers are forcefully buying at higher cost in market at later time. So because of this reason farmers have been moving from villages to cities for the work by thinking agriculture is no more profitable zone.

Disadvantages of Existing system:

- In the existing system all transactions ,dealings of products , purchasing of products were done manually which is time consuming.
- To buy any product user has to collect information about it either by visiting the shop or by asking people.
- There is no computer system for handling payments. All calculations are performed manually which may not always be accurate.

3.2 PROPOSED SYSTEM:

In this portal we are providing direct communication between farmer and vendors. Instead of going from one place to another place. By using this portal farmer can post a product details online to sell their product. And also this help to farmers for storing of food crops by contacting the buyers in advance and choose the plans accordingly. And To eliminate the involvement of intermediate persons in market. To allow farmers in freedom to set the price. The main objective of this project is that there is a direct communication is done in between the User and the farmer.

3.3 SOFTWARE REQUIREMENTS SEPECIFICATION:

After the extensive analysis of the problem in the system we are familiarized with the requirements that the current system needs. The requirement that the system needs is categorized into the functional and non-functional requirements. These requirements are listed below:

3.3.1 FUNCTIONAL REQUIREMENTS:

Functional requirements are the functions or features that must be include in any system to satisfy the business needs and the acceptable to the users. Based in this, the functional requirements that the system must require areas are as follows:

System Modules:

- User
- Staff
- Officer

Module List:

- Farmer
 - Login
 - Create product Add
 - Update/Delete Adds
 - My adds
 - Update application status
 - Logout
- User
 - Register
 - Login
 - Search product
 - My products
 - My profile
 - Logout

3.3.2 NON FUNCTIONAL REQUIREMENTS:

Non functional requirements is a description of features characteristics and attribute of the system as well as any constraints that may limit the boundaries of the proposed system.

The non-functional requirements are essentially based on the performance, information, economy , control and security efficiency and services.

Based on these the non functional requirements are as follows:

- User friendly
- System should provide better accuracy
- To perform with efficient throughput and response time.

4. SYSTEM REQUIREMENTS

4.1 HARDWARE REQUIREMENTS:

- Processor : intel i3
- RAM : 2GB
- Hard disk : Minimum 32GB

4.2 SOFTWARE REQUIREMENTS:

- OPERATING System : Windows
- Front end : CSS ,HTML ,JavaScript
- Backend : Django,MYSQL.

4.3 SOFTWARE DESCRIPTION

In this project the technologies which are used to built the E-MARKET FOR FARMERS are HTML(hypertext markup languages),CSS (cascading style sheet),Bootstrap,javascriptand Django.

HTML [Hyper Text Markup Language]:

- HTML stands for Hyper Text Markup Language.
- HTML describes the structure of web pages using markup.
- HTML elements are the building blocks of HTML pages.
- HTML elements are represented by tags.
- HTML tags label pieces of content such as heading, paragraph, table and so on.
- Browsers do not display the HTML tags, but use them to render the content of the page.

<!doctype html>: The <!doctype> is not an HTML tag. It is an instruction to the web browser about what version of html the pages is written it. This page is written in HTML5 as opposed to say HTML 4.01.

<html>: The <html> element is the root element of an HTML page.

<head>: The <head> element is a container for meta information about the document and it is placed between the <html> and <body> tags.

<title>: The <title> tag is required in all HTML documents and it specifies a title for the document.

<body>: The <body> element contains the visible page content.

<h1>: The <h1> element defines a large heading.

<h6>: The <h6> element defines a small heading.

<p>: The <p> element defines a paragraph.

<a>: The <a> element defines the HTML links. The link's destination is specified in the href attribute.

: The element defines the HTML images. The source file (src), alternative

<button>: The <button> element defines a clickable button, inside a button you can put content.

**
:** The
 element defines the line break. This tag is an empty tag, which means that it has no end tag.

<!-- -->: The <!-- --> element defines the comment tag.

<table>: The <table> element defines the HTML tables.

<tr>: The <tr> element defines the table row.

<th>: The <th> element defines the table header. By default, table headings are bold and centered.

<td>: The <td> element defines the table data/cell.

<div>: The <div> element defines the block level element. A block-level element always starts on a new line and takes up the full width available.

<form>: It defines a form that is used to collect user input.

<label>: The <label> element represents a caption for an item in a user interface.

<input>: The <input> element is the most important form element. The <input> element can be displayed in several ways, depending on the type attribute.

: The element to define an unordered list.

: The element to define an ordered list.

: The element to define a list item.

CSS [Cascading Style Sheet]:

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the colour of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colours are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

CSS can be added to HTML elements in 3 ways:

- **Inline** - by using the style attribute in HTML elements.
- **Internal** - by using a <style> element in the <head> section.
- **External** - by using an external CSS file.

Css Colors: Colors are specified using predefined color names, or RGB, HEX, HSL, RGBA, HSLA values.

- **Css Names:** In HTML, a color can be specified by using a color name.
- **Background Color:** You can set the background color for HTML elements.
- **Text Color:** You can set the color of text.
- **Border Color:** You can set the color of borders.
- **Color Values:** In HTML, colors can also be specified using RGB values, HEX values, HSL values, RGBA values, and HSLA values.

Css Backgrounds: The CSS background properties are used to define the background effects for elements.

- **Background color:** The background-color property specifies the background color of an element.

- **Background Image:** The background-image property specifies an image to use as the background of an element. By default, the image is repeated so it covers the entire element.
- **Background Repeat:** The background-repeat property specifies a background image will be repeated.
- **Background Attachment:** The background-attachment property specifies a background image is fixed or scrolls with the rest of the page.
- **Background Position:** The background-position property specifies a starting position of a background image.

Css Border: The CSS border properties allow you to specify the style, width, and color of an element's border.

- **Border Style:** The Border-style property specifies what kind of border to display.
- **Border Width:** The border-width property specifies the width of the four borders. The width can be set as a specific size (in px, pt, cm, em, etc) or by using one of the three pre-defined values: thin, medium, or thick.
- **Border Color:** The border-color property is used to set the color of the four borders.

Css Margins: The CSS margin properties are used to create space around elements, outside of any defined borders.

- **Margin-Individual Sides:** The CSS margin properties are used to create space around elements, outside of any defined borders.
- **Margin-Shorthand Property:** To shorten the code, it is possible to specify all the margin properties in one property.
- **Margin Collapse:** Top and bottom margins of elements are sometimes collapsed into a single margin that is equal to the largest of the two margins. This does not happen on left and right margins! Only top and bottom margins.

Css Padding: The CSS padding properties are used to generate space around an element's content, inside of any defined borders.

- **Padding-Individual sides:** CSS has properties for specifying the padding for each side of an elements are padding-top, padding-bottom, padding-left, padding-right.
- **Padding - Shorthand Property:** To shorten the code, it is possible to specify all the padding properties in one property.

CSS Height and Width:

- **Setting max-width:** The max-width property is used to set the maximum width of an element. The max-width can be specified in length values, like px, cm, etc., or in percent (%) of the containing block, or set to none (this is default. Means that there is no maximum width).

CSS Box Model: All HTML elements can be considered as boxes. In CSS, the term "box model" is used when talking about design and layout.

The CSS box model is essentially a box that wraps around every HTML element. It consists of margins, borders, padding, and the actual content.

- **Content** - The content of the box, where text and images appear
- **Padding** - Clears an area around the content. The padding is transparent
- **Border** - A border that goes around the padding and content
- **Margin** - Clears an area outside the border. The margin is transparent

The box model allows us to add a border around elements, and to define space between elements.

CSS Outline: An outline is a line that is drawn around elements, OUTSIDE the borders, to make the element "stand out".

- **Outline Style:** The outline-style property specifies the style of the outline, and can have one of the following values like dotted, dashed, solid, double etc.
- **Outline Color:** The outline-color property is used to set the color of the outline.
- **Outline Width:** The outline-width property specifies the width of the outline, and can have one of the following values:
 - thin (typically 1px)
 - medium (typically 3px)

- thick (typically 5px)
- A specific size (in px, pt, cm, em, etc)
- **Outline Offset:** The outline-offset property adds space between an outline and the edge/border of an element. The space between an element and its outline is transparent.

CSS Text:

- **Text Color:** The color property is used to set the color of the text.
- **Text Alignment:** The text-align property is used to set the horizontal alignment of a text. A text can be left or right aligned, centered, or justified.
- **Text Decoration:** The text-decoration property is used to set or remove decorations from text. The value text-decoration: none; is often used to remove underlines from link.
- **Text Transformation:** The text-transform property is used to specify uppercase and lowercase letters in a text. It can be used to turn everything into uppercase or lowercase letters, or capitalize the first letter of each word.
- **Text Indentation:** The text-indent property is used to specify the indentation of the first line of a text.
- **Letter Spacing:** The letter-spacing property is used to specify the space between the characters in a text.
- **Line Height:** The line-height property is used to specify the space between lines.
- **Text Direction:** The direction property is used to change the text direction of an element.
- **Word Spacing:** The word-spacing property is used to specify the space between the words in a text.

CSS Font Families:

In CSS, there are two types of font family names:

- **generic family** - a group of font families with a similar look (like "Serif" or "Monospace")

- **font family** - a specific font family (like "Times New Roman" or "Arial"). The font family of a text is set with the font-family property. The font-family property should hold several font names as a "fallback" system. If the browser does not support the first font, it tries the next font, and so on.
- **Font Style:** The font-style property is mostly used to specify italic text. This property has three values:
 - **normal** - The text is shown normally.
 - **italic** - The text is shown in italics.
 - **oblique** - The text is "leaning" (oblique is very similar to italic, but less supported)
- **Font Size:** The font-size property sets the size of the text. Being able to manage the text size is important in web design. However, you should not use font size adjustments to make paragraphs look like headings, or headings look like paragraphs.

Css Links: Links can be styled differently depending on what **state** they are in.

The four links states are:

- a:link - a normal, unvisited link
- a:visited - a link the user has visited
- a:hover - a link when the user mouses over it
- a:active - a link the moment it is clicked

JAVA SCRIPT:

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

- JavaScript is a lightweight, interpreted programming language.
- Designed for creating network-centric applications.
- Complementary to and integrated with Java.
- Complementary to and integrated with HTML.
- Open and cross-platform.

JavaScript Can Change HTML Content: One of many JavaScript HTML methods is `getElementById()`.

<script>Tag: In HTML, JavaScript code must be inserted between `<script>` and `</script>` tags.

JavaScript in <head> or <body>: You can place any number of scripts in an HTML document. Scripts can be placed in the `<body>`, or in the `<head>` section of an HTML page, or in both.

External JavaScript:

External scripts are practical when the same code is used in many different web pages.

JavaScript files have the file extension **.js**.

To use an external script, put the name of the script file in the `src` (source) attribute of a `<script>` tag:

JavaScript Display Possibilities:

JavaScript can "display" data in different ways:

- Writing into an HTML element, using `innerHTML`.
- Writing into the HTML output using `document.write()`.
- Writing into an alert box, using `window.alert()`.
- Writing into the browser console, using `console.log()`.

Using innerHTML:

To access an HTML element, JavaScript can use the `document.getElementById(id)` method. The `id` attribute defines the HTML element. The `innerHTML` property defines the HTML content.

Using `window.alert()`: You can use an alert box to display data.

Using `console.log()`: For debugging purposes, you can use the `console.log()` method to display data.

JavaScript White Space: JavaScript ignores multiple spaces. You can add white space to your script to make it more readable.

JavaScript Code Blocks: JavaScript statements can be grouped together in code blocks, inside curly brackets { ... }. The purpose of code blocks is to define statements to be executed together.

JavaScript Values: The JavaScript syntax defines two types of values: Fixed values and variable values. Fixed values are called **literals**. Variable values are called **variables**.

JavaScript Variables: In a programming language, **variables** are used to **store** data values. JavaScript uses the var keyword to **declare** variables. An **equal sign** is used to **assign values** to variables.

JavaScript Identifiers:

All JavaScript **variables** must be **identified** with **unique names**.

These unique names are called **identifiers**.

Identifiers can be short names (like x and y) or more descriptive names (age, sum, totalVolume).

The general rules for constructing names for variables (unique identifiers) are:

- Names can contain letters, digits, underscores, and dollar signs.
- Names must begin with a letter
- Names can also begin with \$ and _ (but we will not use it in this tutorial)
- Names are case sensitive (y and Y are different variables)
- Reserved words (like JavaScript keywords) cannot be used as names

JavaScript Functions:

A JavaScript function is a block of code designed to perform a particular task.

A JavaScript function is executed when "something" invokes it.

Function Invocation:

The code inside the function will execute when "something" **invokes** (calls) the function:

- When an event occurs (when a user clicks a button)
- When it is invoked (called) from JavaScript code
- Automatically (self invoked)

Function Return:

When JavaScript reaches a return statement, the function will stop executing.

If the function was invoked from a statement, JavaScript will "return" to execute the code after the invoking statement.

JavaScript Events:

HTML events are "**things**" that happen to HTML elements.

When JavaScript is used in HTML pages, JavaScript can "**react**" on these events.

HTML Events:

An HTML event can be something the browser does, or something a user does.

Here are some examples of HTML events:

- An HTML web page has finished loading
- An HTML input field was changed
- An HTML button was clicked

Often, when events happen, you may want to do something. JavaScript lets you execute code when events are detected.

HTML allows event handler attributes, **with JavaScript code**, to be added to HTML elements.

- **Onchange-** An HTML element has been changed.
- **Onclick-** The user clicks an HTML element.
- **Onmouseover-** The user moves the mouse over an HTML element.
- **Onmouseout-** The user moves the mouse away from an HTML element.
- **Onkeydown-** The user pushes a keyboard key.
- **Onload-** The browser has finished loading the page.

BOOTSTRAP:

Bootstrap is a free and open source front end development framework for the creation of websites and web apps. The Bootstrap framework is built on HTML, CSS and JavaScript (JS) to facilitate the development of responsive.

.align : A set of utility classes that are equivalent to writing the css property. You can use this on inline and table cell elements.

.align-content : Added to the parent flexbox container to determining how the elements are aligned horizontally.

.align-items : Class added to flexbox child items to specify if it should align towards the top or bottom of the container (start, end).

.bg : Background color utility classes.

.border : A versatile border utility class that lets you add/remove borders on a side or change a border color.

.btn-outline: A button variation to have outlined buttons instead of a solid background.

.carousel-control : When you have an image carousel with pagination you will use this class on the previous and next anchor links.

.carousel-fade : Animates the slide transition with a crossfade instead of a slide.

.form-control-plaintext : Use the class to remove the default form field styling and preserve the correct margin and padding.

.form-row : Works similar to a grid. but is more compact to make the form look more uniform.

.h-* : Height utility class that makes the element a percentage height of its parent element.

.justify-content-*-* : Class specifies where the flex items will be positioned inside the container.

.nav-fill : Makes all nav items use all available horizontal space. Nav items are different widths based on their content.

.nav-justified : Makes all nav items equal width and use all available horizontal space.

.navbar-collapse : The nav links that are collapsed and shown when toggled on mobile widths.

.navbar-expand-* : Since the navbar is displayed collapse on mobile first, this class specifies what breakpoint you want the navbar to not be collapsed.

.navbar-text : Vertically centres text inside a navbar.

.navbar-toggler-icon : The cheeseburger navigation icon is set using an svg background image of three horizontal lines.

.btn-group-lg : Increases the default button group size.

.btn-group-sm : Decreases the default button group size.

.btn-group-toggle : This class replaces an input checkbox with a custom style that is toggleable on click.

.btn-outline-* : Transparent background with colored text and border. In bootstrap there are some button styles. There are

.btn,.btn-default,.btn-primary,.btn-success,.btn-info,.btn-warning,.btn-danger,.btn-link.

.carousel-item : The wrapper class applied to each individual carousel item.

.col-form-label : Class added to form labels to apply consistent padding and margins

.display-* : This set of classes increases the font size of headings in 4 stages. These classes are used for headings outside of the main content of the page like jumbotrons and page headers. Append (1-4) to the end to adjust size.

.dropdown-item : This class is added to each link item shown in a dropdown menu.

.dropdown-toggle-split : Removes the interactivity from a dropdown so it does not appear clickable.

.font-* : italic, weight-bold, weight-light, weight-normal, monospace.

.form-inline : Use this class to have a series of labels and form elements on a single horizontal row.

.form-text : This class is used for help text alongside form elements. You can add .text-muted to make the text lighter in color.

.input-group-text : This class adds the background color and text styles to the text inside an input group.

.nav-item : If your nav uses a list add this class to each list item for the proper spacing.

.nav-link : Each anchor link inside your nav is given this class in order to have the proper styling.

.nav-pills : Use this class along with .nav to make each nav link into a button.

.navbar-brand : Most navbars contain a logo or brand. This class is added to the anchor.

.navbar : Navigation header class.

.navbar-nav : The wrapper class of the navigation elements excluding the brand.

.navbar-light : Add this class to your navbar if you would like it to have a light background and dark text.

.navbar-toggler : The infamous cheeseburger icon to signify a navigation menu on mobile.

.position-* : Not responsive, but a group of utility classes to add common position values. Position may be absolute, fixed, relative, static, sticky.

.container : Fixed width container with widths determined by screen sizes. Equal margin on the left and right

.container-fluid : Spans the full width of the screen.

.media : Media components are image heading and description text items. Blog comments, portfolio projects, album covers, etc.

.modal : The parent wrapper class of modal content.

.modal-body : The modal body content classes : Header - Body – Footer.

.modal-content : modal-content contains modal-body, modal-header, and modal-footer.

.modal-dialog : The secondary wrapper class of the entire modal content.

.modal-header : The header section of the modal that contains the title and close button.

.modal-footer : The footer of the modal that contains action buttons or help text.

.tooltip : The Tooltip plugin is small pop-up box that appears when the user moves the mouse pointer over an element.

.popover : It is a pop-up box that appears when the user clicks on an element. The difference is that the popover can contain much more content.

.panel : A panel in bootstrap is a bordered box with some padding around its content. To color the panel, use contextual classes `.panel-default`, `.panel-primary`, `.panel-success`, `.panel-info`, `.panel-warning`, or `.panel-danger`.

.pagination : If you have a web site with lots of pages, you may wish to add some sort of pagination to each page.

. **Jumbotron:** A jumbotron indicates a big box for calling extra attention to some special content or information.

A jumbotron is displayed as a grey box with rounded corners. It also enlarges the font sizes of the text inside it.

.**pager:** Pager is also a form of pagination,Pager provides previous and next buttons (links).

. **Glyphicons:** Glyphicons can be used in text, buttons, toolbars, navigation, forms.Examples of glyphicons are Envelope glyphicon,Print glyphicon, Search glyphicon ,Download glyphicon.

5. MODULE DESCRIPTION

MODULES:

Registration:

This module acts as an main window for this applications which provides the entry for this applications system. From this module we can register the register the requests of user and farmer.

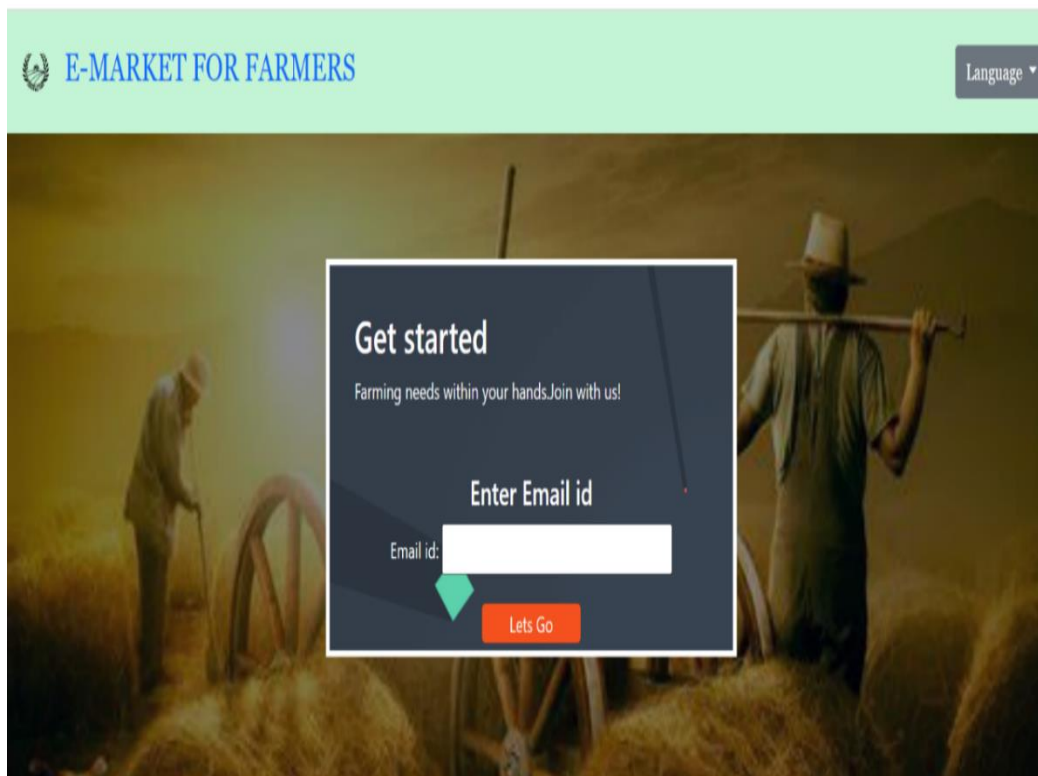
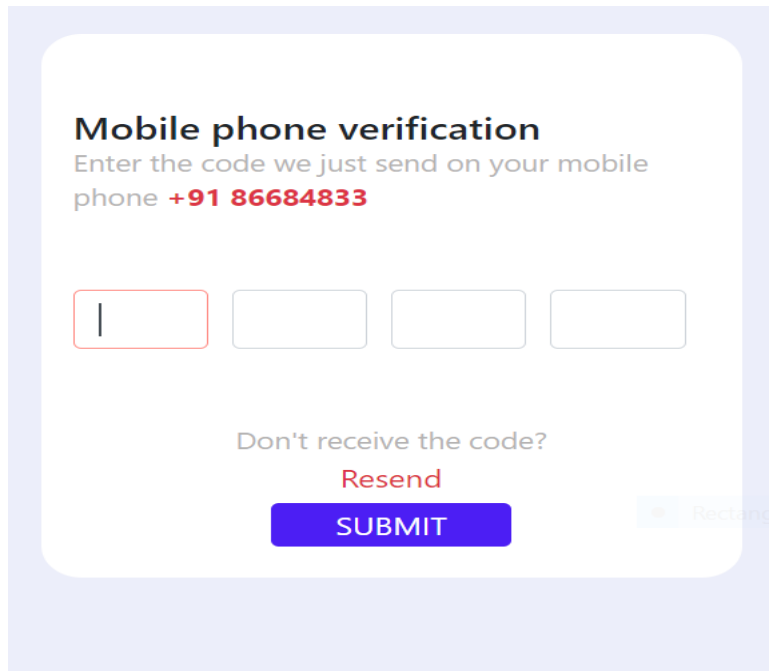


Fig:5.1Registration page

Verification page:

A mobile phone verification form with a light blue background. The form is a white rounded rectangle containing the title "Mobile phone verification", a prompt to enter a code, a phone number, four input fields for the code, a "Resend" link, and a "SUBMIT" button.

Mobile phone verification
Enter the code we just send on your mobile
phone **+91 86684833**

Don't receive the code?
[Resend](#)

SUBMIT

This module acts as the entering of identifier information into a system by a user in order to access that system by providing mobile verification code .if it is valid mobile number then the user control is directed to actual page of e-market for farmers.

6.SYSTEM DESIGN

6.1 SYSTEM ARCHITECTURE:

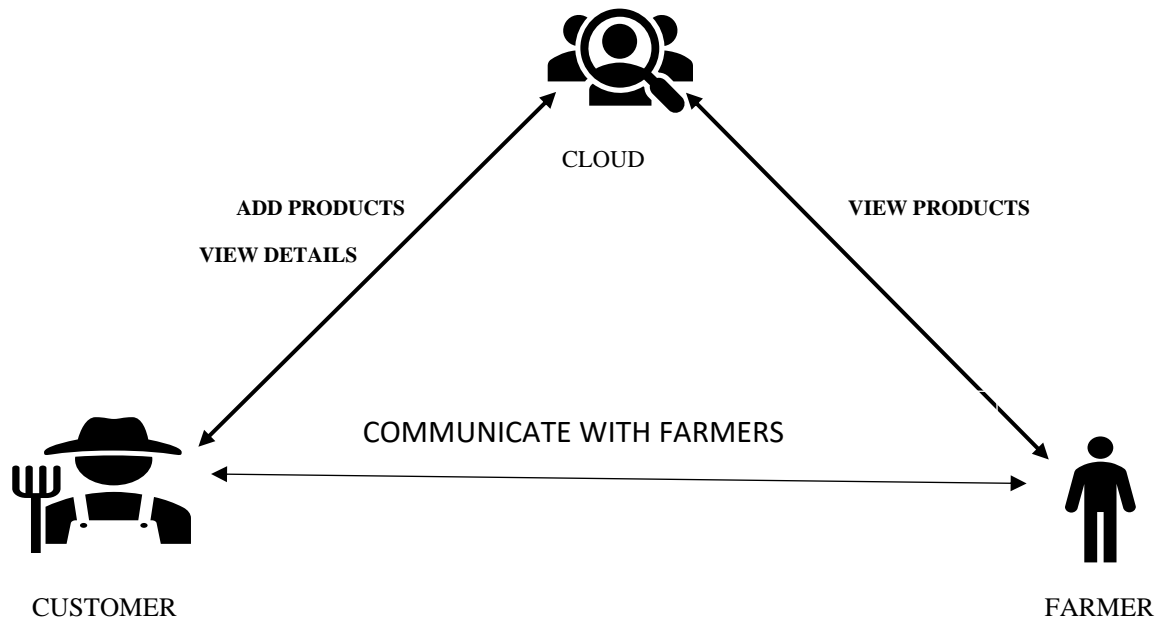


Fig: 7.1 SYSTEM ARCHIETECTURE

In this project there are two major roles are there one is farmer and another one is customer.If the farmer wants to sell their products they can login and add products to the cloud (i.e.,into this portal).and If the customer want to buy a product they can see product and farmer details in this portal and communicate with the farmers.

6.2 Introduction to UML diagrams:

UML Diagrams:

UML is a method for describing the system architecture in detail using blueprint.

- UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.
- UML is a very important part of developing objects oriented software and the software development process.
- UML uses mostly graphical notations to express the design of software projects.

Definition:

UML is a general-purpose visual modeling language that is used to specify visualize, construct and document the artifacts of the software system.

UML is a Language:

Specifying means building models that are precise, unambiguous and complete. In particular, the UML address the specification of all the important analysis, design and implementation decisions that must be made in developing and displaying a software intensive system.

UML Visualization:

The UML includes both graphical and textual representation. It makes easy to visualize the system and for better understanding.

UML constructing:

UML, models can be directly connected to a variety of programming languages and it is sufficiently expressive and free from any ambiguity to permit the direct execution of models.

UML Documenting:

UML provides variety of documents in addition raw executable codes.

- The use case view of a system encompasses the use cases that describe the behavior of the system as seen by its end users, analysts and testers.
- The design view of a system encompasses the classes, interfaces and collaborations that form the vocabulary of the problem and its solution.
- The process view of a system encompasses the threads and processes that form the system's concurrency and synchronization mechanism.
- The implementation view of a system encompasses the components and files that are used to assemble and releases the physical system .

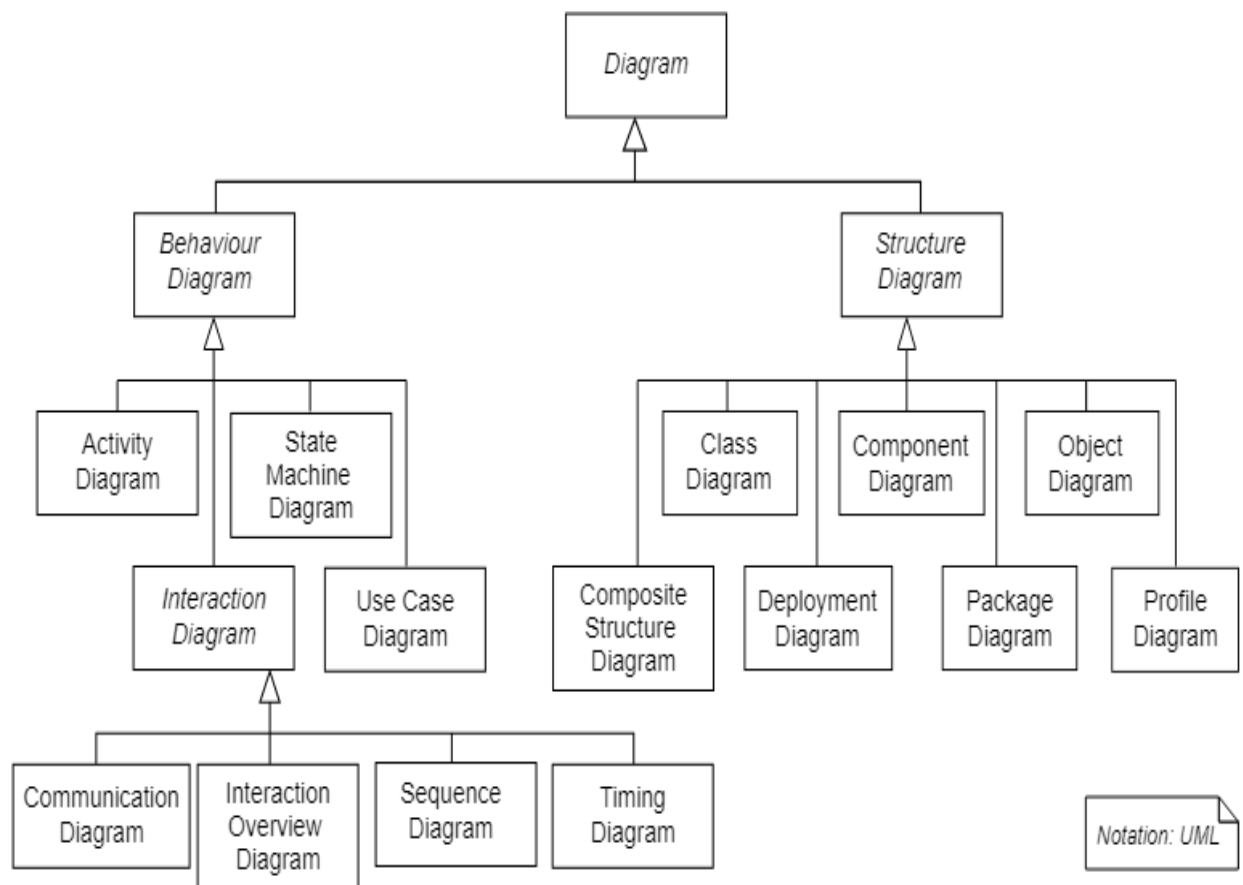
Uses of UML:

The UML is intended primarily for software intensive systems. It has been used effectively for such domain as

- Enterprise information system

- Banking and financial service
- Telecommunication
- Scientific fields
- Transportation
- Retails
- Defence
- Medical electronics
- Distributed web

TYPES OF UML DIAGRAMS:



1. Structural Diagrams

Structural diagrams depict a static view or structure of a system. It is widely used in the documentation of software architecture. It embraces class diagrams, composite structure diagrams, component diagrams, deployment diagrams, object diagrams, and package diagrams. It presents an outline for the system. It stresses the elements to be present that are to be modeled.

- **Class Diagram:** Class diagrams are one of the most widely used diagrams. It is the backbone of all the object-oriented software systems. It depicts the static structure of the system. It displays the system's class, attributes, and methods. It is helpful in recognizing the relation between different objects as well as classes.
- **Composite Structure Diagram:** The composite structure diagrams show parts within the class. It displays the relationship between the parts and their configuration that ascertain the behavior of the class. It makes full use of ports, parts, and connectors to portray the internal structure of a structured classifier. It is similar to class diagrams, just the fact it represents individual parts in a detailed manner when compared with class diagrams.
- **Object Diagram:** It describes the static structure of a system at a particular point in time. It can be used to test the accuracy of class diagrams. It represents distinct instances of classes and the relationship between them at a time.
- **Component Diagram:** It portrays the organization of the physical components within the system. It is used for modeling execution details. It determines whether the desired functional requirements have been considered by the planned development or not, as it depicts the structural relationships between the elements of a software system.
- **Deployment Diagram:** It presents the system's software and its hardware by telling what the existing physical components are and what software components are running on them. It produces information about system software. It is incorporated whenever software is used, distributed, or deployed across multiple machines with dissimilar configurations.
- **Package Diagram:** It is used to illustrate how the packages and their elements are organized. It shows the dependencies between distinct packages. It manages UML diagrams by making it easily understandable. It is used for organizing the class and use case diagrams.

2. Behavioral Diagrams:

Behavioral diagrams portray a dynamic view of a system or the behavior of a system, which describes the functioning of the system. It includes use case diagrams, state diagrams, and activity diagrams. It defines the interaction within the system.

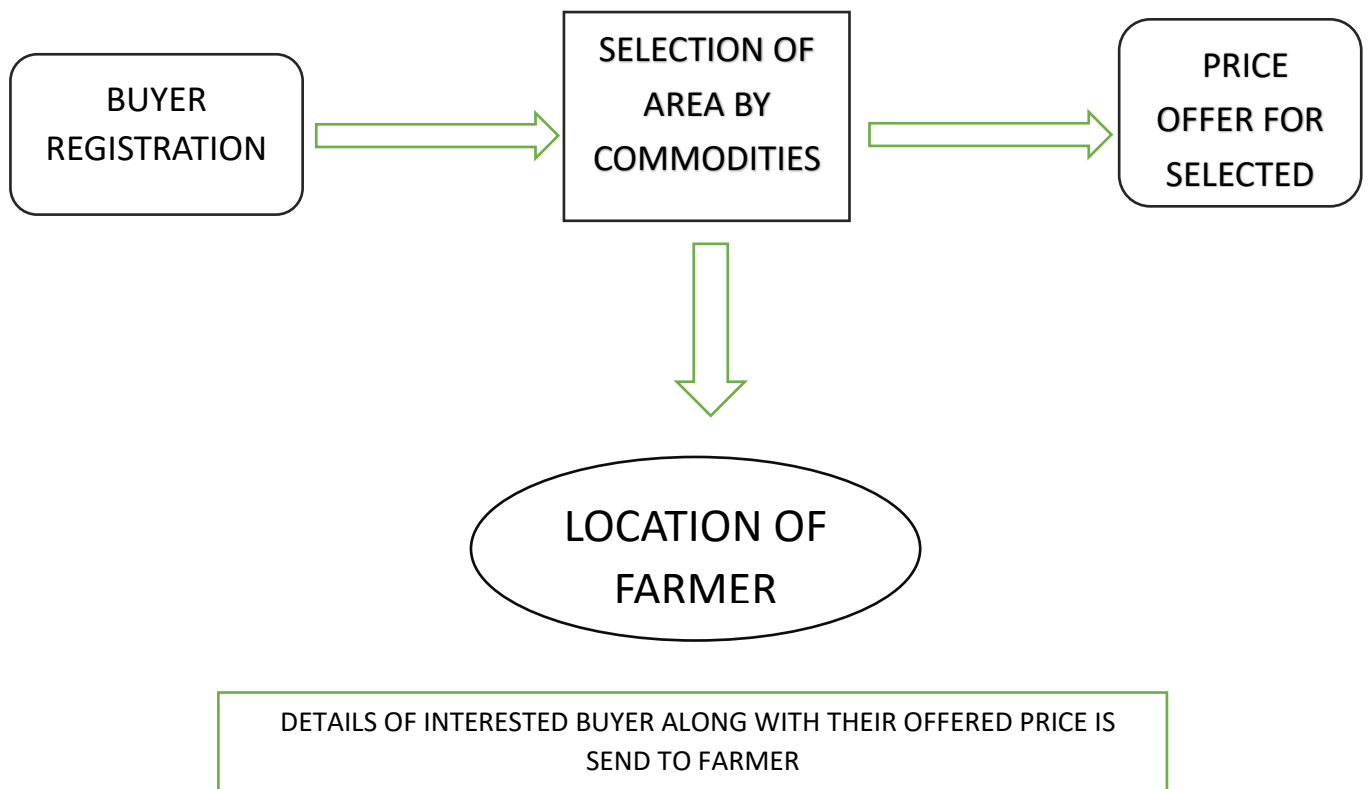
- **State Machine Diagram:** It is a behavioral diagram. it portrays the system's behavior utilizing finite state transitions. It is also known as the **State-charts** diagram. It models the dynamic behavior of a class in response to external stimuli.
- **Activity Diagram:** It models the flow of control from one activity to the other. With the help of an activity diagram, we can model sequential and concurrent activities. It visually depicts the workflow as well as what causes an event to occur.
- **Use Case Diagram:** It represents the functionality of a system by utilizing actors and use cases. It encapsulates the functional requirement of a system and its association with actors. It portrays the use case view of a system.

3. Interaction Diagrams

Interaction diagrams are a subclass of behavioral diagrams that give emphasis to object interactions and also depicts the flow between various use case elements of a system. In simple words, it shows how objects interact with each other and how the data flows within them. It consists of communication, interaction overview, sequence, and timing diagrams.

- **Sequence Diagram:** It shows the interactions between the objects in terms of messages exchanged over time. It delineates in what order and how the object functions are in a system.
- **Communication Diagram:** It shows the interchange of sequence messages between the objects. It focuses on objects and their relations. It describes the static and dynamic behavior of a system.
- **Timing Diagram:** It is a special kind of sequence diagram used to depict the object's behavior over a specific period of time. It governs the change in state and object behavior by showing the time and duration constraints.
- **Interaction Overview diagram:** It is a mixture of activity and sequence diagram that depicts a sequence of actions to simplify the complex interactions into simple interactions

6.3 CO-ORDINATION BETWEEN SELLER AND THE FARMER



6.3 IMPLEMENTATION:

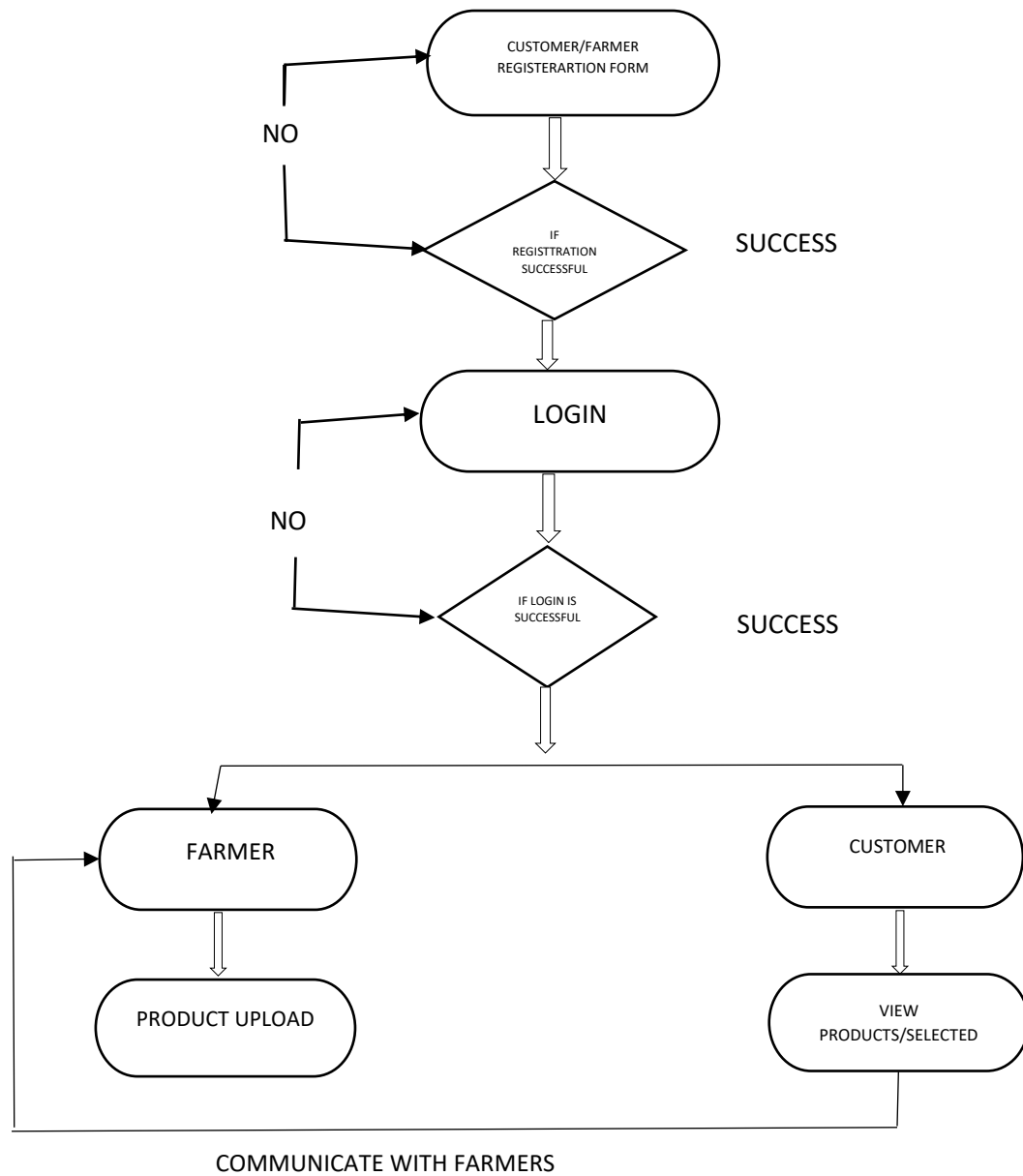


Fig:6.3 IMPLEMENTATION

6.4 UML DIAGRAMS

6.4.1 USE CASE DIAGRAM

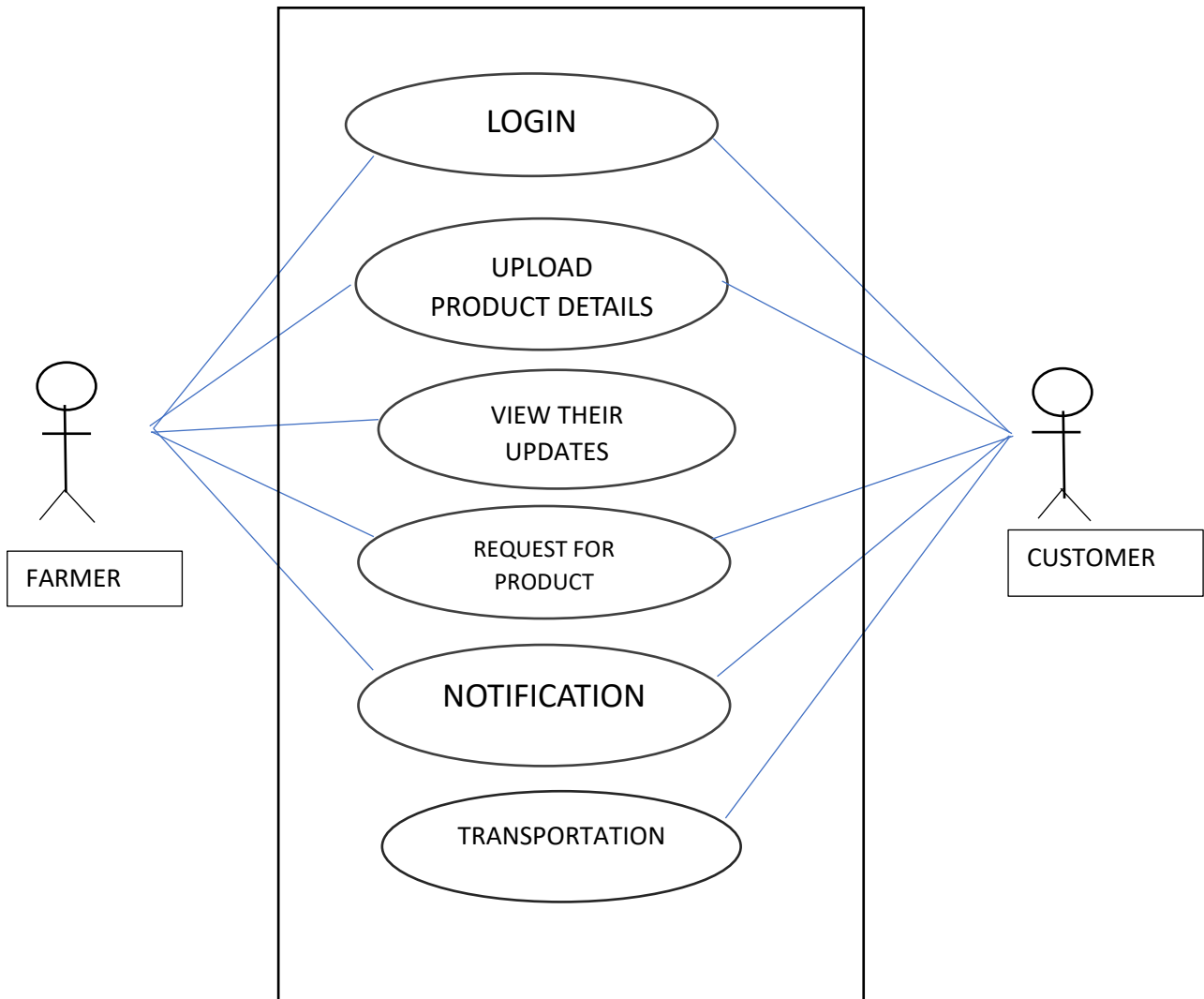


Fig:6.4.1 USE CASE DIAGRAM

6.4.2 CLASS DIAGRAM :

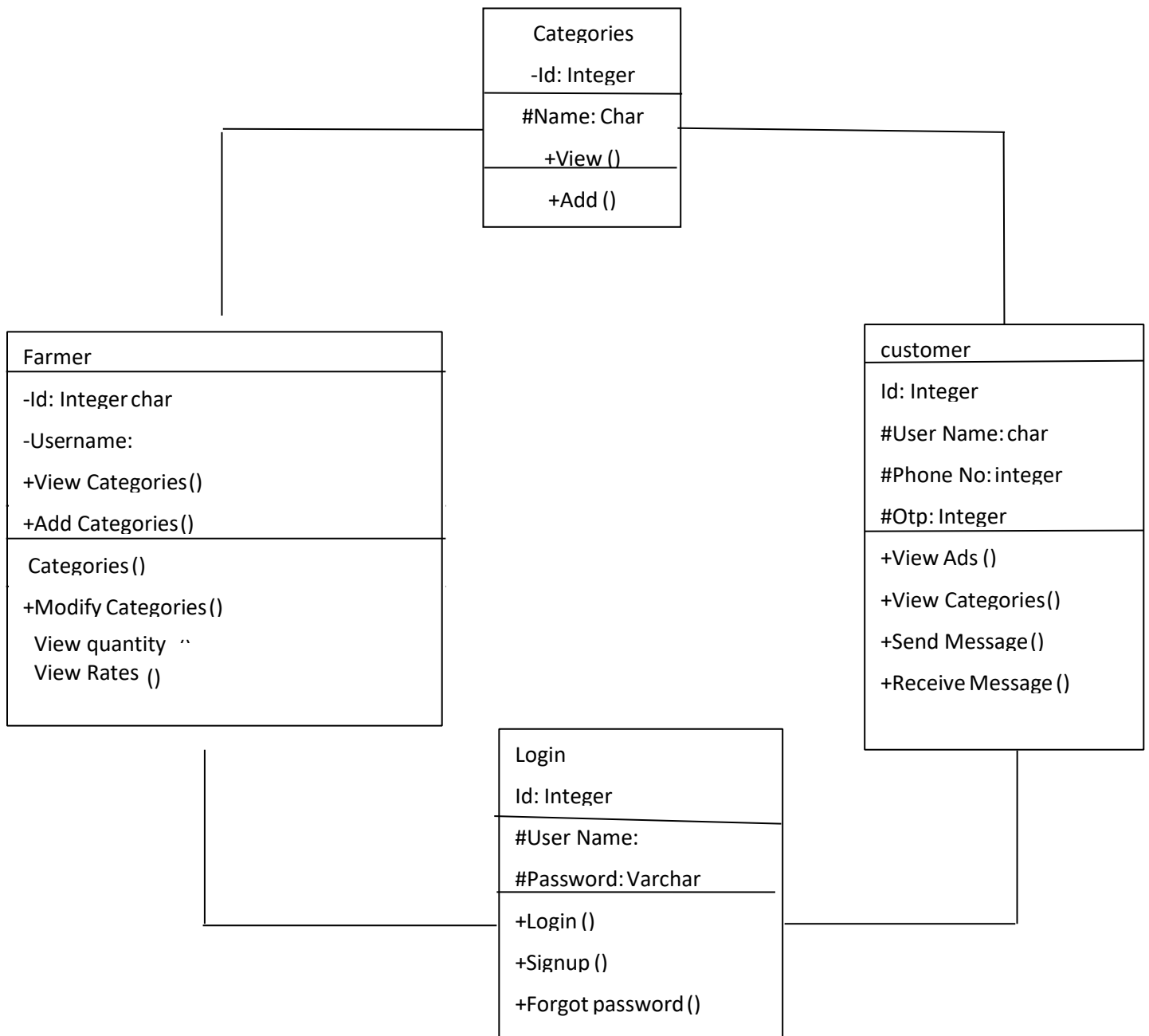


Fig:6.4.2 CLASS DIAGRAM

6.4.3 SEQUENTIAL DIAGRAM:

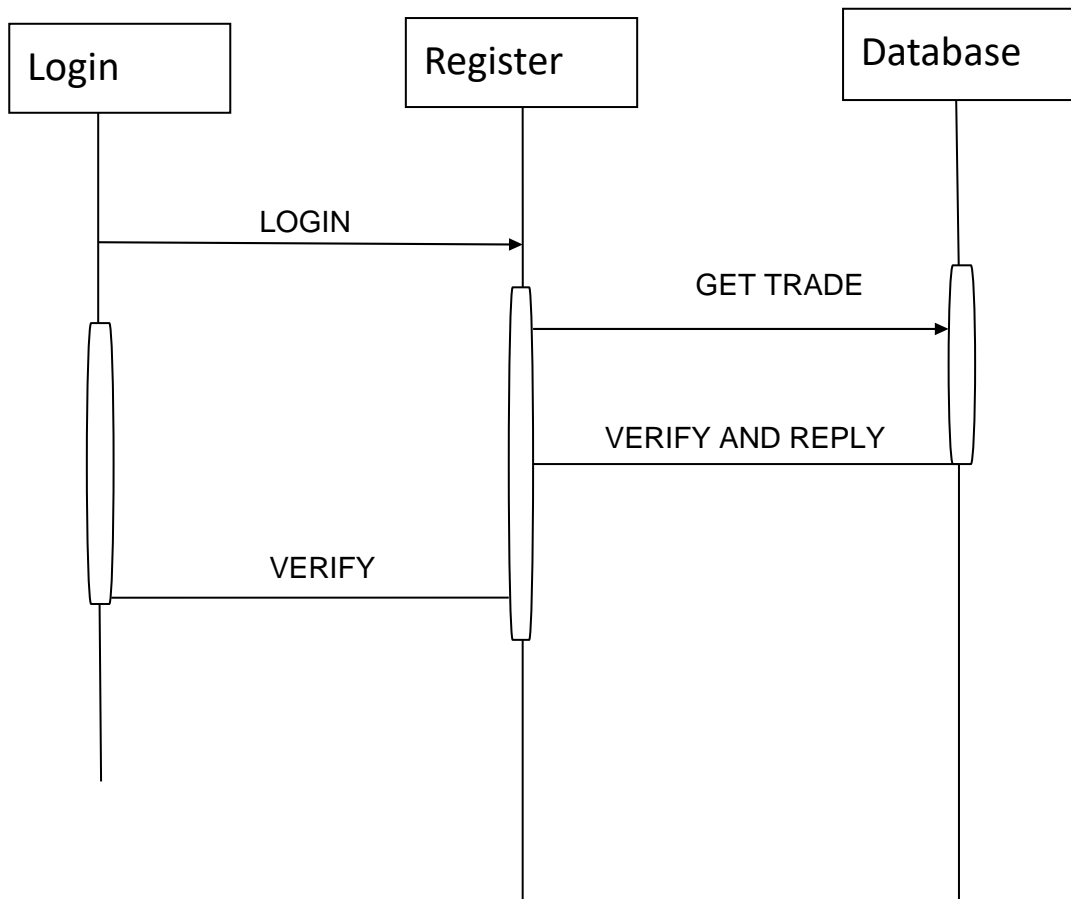


Fig:6.4.3 SEQUENTIAL DIAGRAM

6.4.4 COMPONENT DIAGRAM:

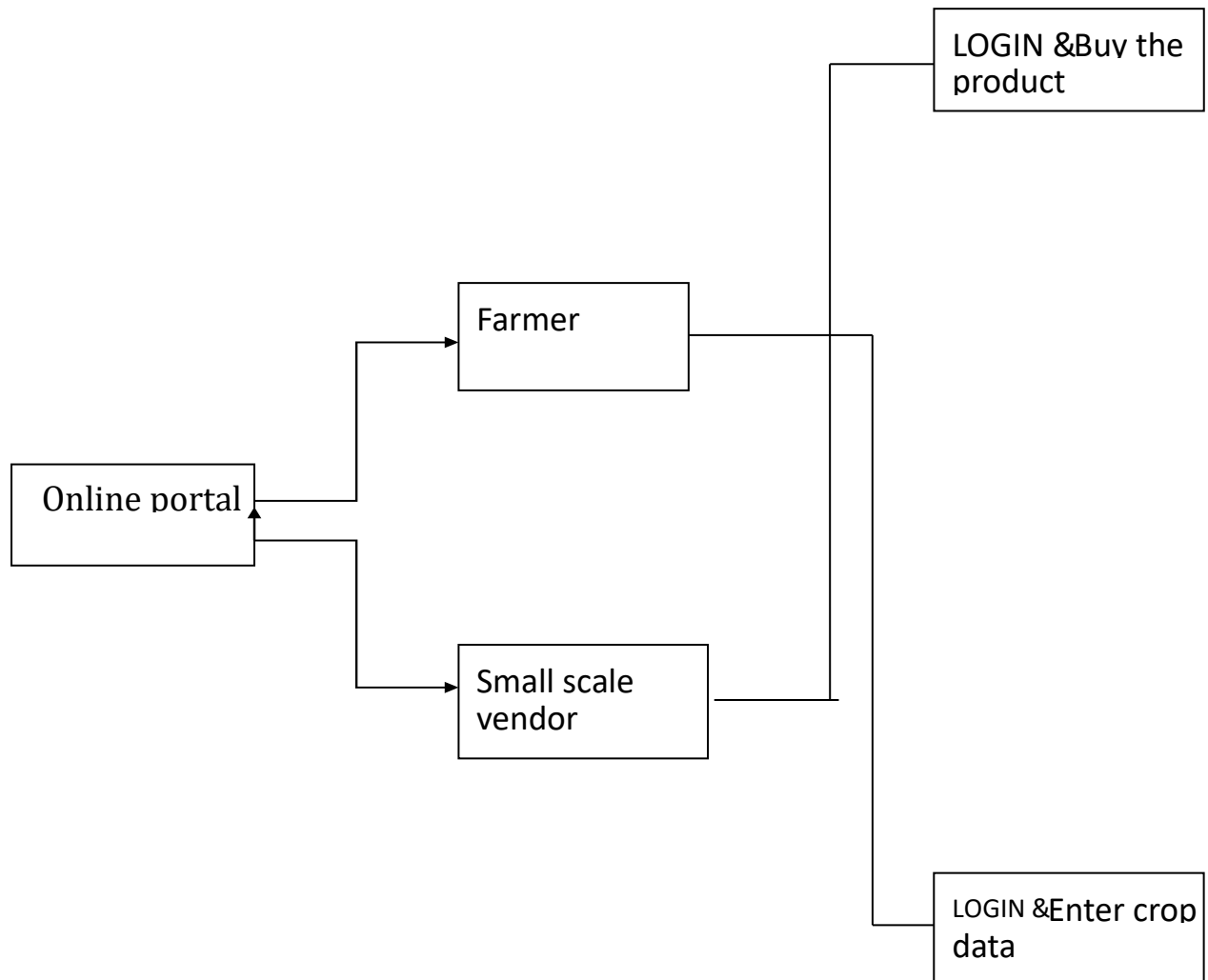


Fig:6.4.4 COMPONENT DIAGRAM

6.4.5 DEPLOYMENT DIAGRAM:

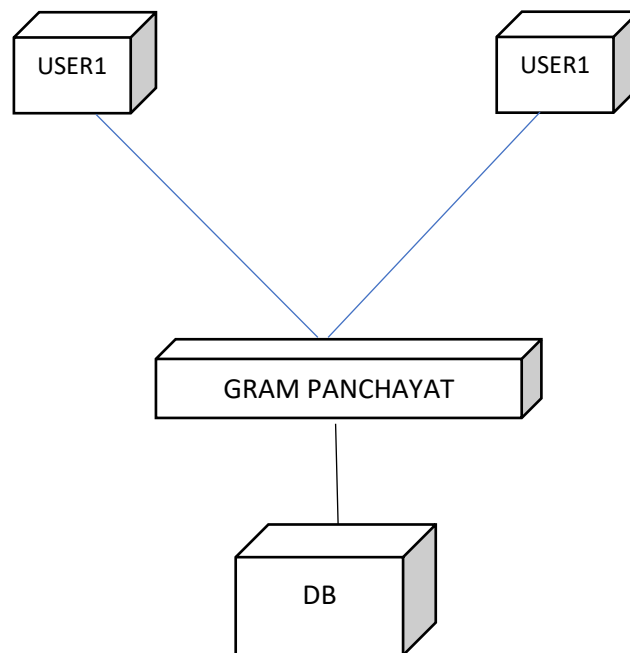


Fig:6.4.5 DEPLOYMENT DIAGRAM

7. CODING

INDEX.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>E-MARKET FOR FARMERS</title>
  <link rel="stylesheet" href="f.css">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">

  <!-- Latest compiled JavaScript -->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
  <nav class="navbar ">
    <div class="container-fluid">
      <a class="navbar-brand" href="#">
        <div class="nav-brand d-flex ">
          
          <h3>E-MARKET FOR FARMERS</h3>
        </div>
      </a>
      <div class="dropdown">
        <button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton2" data-bs-toggle="dropdown" aria-expanded="false">
          Language
        </button>
        <ul class="dropdown-menu dropdown-menu-light" aria-labelledby="dropdownMenuButton2">
          <li><a class="dropdown-item active" href="#">English</a></li>
          <li><a class="dropdown-item" href="#">Telugu</a></li>
        </ul>
      </div>
    </div>
  </nav>

  <div class="bigcontainer">
    <div class="bg-overlay"></div>
    <div class="maincont">
      <div class="getstarted">
        <h2>Get started </h2>
      </div>
    </div>
  </div>
</body>
</html>
```

```

        <p>Farming needs within your hands.Join with us!</p>
    </div>

    <div class="emailcontain">
        <h4><center>Enter Email id </center></h4>
        <center><p>Email id: <input type="email" name="" id=""></p></center>
    </div>

    <center>
        <a href="login.html" ><button onclick="" class="letsgo">Lets Go </button></a>
    </center>
</div>
</div>

</body>
</html>

```

INDEX.CSS

```

*{
    margin:0%;
    padding:0%;
    box-sizing: border-box;
}
body,html {
    overflow-x: hidden;
    font-family: sans-serif;
    font-size: 16px;
}

.navbar
{
    background-color: rgb(196, 244, 214);
    display:flex;
    font-family: Georgia, 'Times New Roman', Times, serif;
}

.container-fluid a{
    align-items: center;
}

.container-fluid img{
    border-radius: 50%;
}

h3{
    padding:10px;
}

.d-flex {

```

```

    display: flex;
}
.nav-brand{
    align-items: center;
    justify-content: center;
}
.bigcontainer{
    background-image:
url('https://th.bing.com/th/id/OIP.ZE1_jfU1CLQGAtrw00AbXgHaF7?pid=ImgDet&rs=1');
    background-color: #cccccc;
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;
    padding: 20px;
    min-height: 80vh;
    display: flex;
    justify-content: center;
    position: relative;
}

.maincont{
    background-image: url('http://csshint.com/wp-content/uploads/2020/01/CSS-Animated-Backgrounds.gif');
    margin:auto 80px ;
    /* height: 33%; */
    width: 500px;
    color: white;
    border:white solid 5px;
    z-index: 2;
}

.getstarted{

    color: white;
    padding: 30px;
}
input{

    width: 280px;
    padding: 04px 12px;

}
.letsgo{
    display: inline-block;
    border-radius: 4px;
    background-color: #f4511e;
    border: none;
    color: #FFFFFF;
    text-align: center;

```

```
padding: 2px;
width: 120px;
transition: all 0.5s;
cursor: pointer;
margin: 5px;
}
.bg-overlay {
position: absolute;
top: 0;
left: 0;
width: 100%;
height: 100%;
background: rgba(1, 0, 12, 0.461);
}
button:hover{
    box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
```

LOGIN.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">

<!-- Latest compiled JavaScript -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
    <link rel="stylesheet" href="login.css">
    <title>login form</title>
</head>
<body>
    <div class="d-flex justify-content-center align-items-center container">
        <div class="card py-5 px-3">
            <h5 class="m-0">Mobile phone verification</h5><span class="mobile-text">Enter the code we
just send on your mobile phone <b class="text-danger">+91 86684833</b></span>
            <div class="d-flex flex-row mt-5"><input type="text" class="form-control" autofocus=""><input
type="text" class="form-control"><input type="text" class="form-control"><input type="text"
class="form-control"></div>
            <div class="text-center mt-5"><span class="d-block mobile-text">Don't receive the
code?</span><span class="font-weight-bold text-danger cursor">Resend</span></div>
```

```

        <center> <a href="profile.html" ><button
onclick="" class="submit">SUBMIT</button></a></center>

    </div>
</div>
</body>
</html>

```

LOGIN.CSS

```

.card {
    width: 350px;
    padding: 10px;
    border-radius: 20px;
    background: #fff;
    border: none;
    height: 350px;
    position: relative
}

.container {
    height: 100vh
}

body {
    background: #eceefa;
}

.mobile-text {
    color: #989696b8;
    font-size: 15px
}

.form-control {
    margin-right: 12px
}

.form-control:focus {
    color: #495057;
    background-color: #fff;
    border-color: #ff8880;
    outline: 0;
    box-shadow: none
}

.cursor {
    cursor: pointer
}

```



```
.submit{
  display: inline-block;
  border-radius: 4px;
  background-color: #4c1ef4;
  border: none;
  color: #ffffff;
  text-align: center;
  padding: 2px;
  width: 120px;
  transition: all 0.5s;
  cursor: pointer;
  margin: 5px;
}
```

PROFILE.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="profile.css">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
  integrity="sha384-
  EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOMLASjC"
  crossorigin="anonymous">
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/font-awesome@6.0.0/css/all.min.css"
  />
  <title> profile</title>
</head>
<body>

  <center>
  <div class="cont">
    <form>
      <!-- <fieldset disabled></fieldset> -->
      <legend><i class="fa-solid fa-circle-user"></i></legend>
      <div class="mb-3">
        <label for="disabledTextInput" class="form-label">Full name :</label>
        <input type="text" id="disabledTextInput" class="form-control" placeholder="enter name">
      </div>
      <div class="mb-3">
        <label for="disabledTextInput" class="form-label">Mobile Number :</label>
        <input type="number" id="disabledTextInput" class="form-control" placeholder="Enter mobile
number">
      </div>
      <div class="mb-3">
```

```

        <label for="disabledTextInput" class="form-label">Address :</label>
        <input type="text" id="disabledTextInput" class="form-control" placeholder="Enter address">
    </div>
    <div class="mb-3">
    </div>
    <a href="home.html" class="btn btn-primary">upload</a>

    <!-- <a href="home.html" ><button onclick="" class="submit">upload</button></a> -->
    <!-- </fieldset> -->
</form>

</div>
</center>
</body>
</html>

```

PROFILE.CSS

```

*
{
    margin:0%;
    padding:0%;
}
.cont
{
    margin-top: 30px;
    display:flex;
    border: rgb(90, 61, 249) solid 5px;
    height: 404px;
    width: 28%;
    align-items: center;
    justify-content: center;
    background-color:rgb(169, 169, 247);

}
i {
    font-size: 38px;
}
.cont form #icon{
    size: 30px;
}

```

HOME.HTML

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOMLASjC"
crossorigin="anonymous">
  <title>home</title>

  <style>
    *{
margin:0%;
padding:0%;
}
    body,html {
      overflow-x: hidden;
      font-family: sans-serif;
      font-size: 16px;
    }
    .back-video
    {
      position:absolute;

      width: 100%;
      /* right:0; */
      /* bottom:1; */
      /* z-index: index -1; */

    }
    .navbar
    {
      background-color: rgb(158, 247, 192);
      display:flex;
      font-family: Georgia, 'Times New Roman', Times, serif;
    }

    h3{
      padding:10px;
    }
    #actions {
```

```

    background-
image:url("https://dm0qx8t0i9gc9.cloudfront.net/watermarks/video/V7hI1imKeijyy0rx1/videoblocks-
beautiful-farm-landscape-rural-aerial-drone-shot-at-
sunset_sbzs6zey8h__de2eb75ec81aac67347abb3b1d9e3459__P360.mp4");
    background-size: cover;
    padding: 0px 0;
}
.container
{
    width:80%;
    margin: auto;

    /* padding: 30px; */
}

.row {
    display: flex;
    flex-wrap: wrap;
    gap: 30px;
    align-items: center;
    justify-content: center;
}
.card {
    width: 300px;
    padding: 0;
    margin-top: 30px;
}

.card img {
    max-width: 100%;
    height: 200px;
}
@media (min-aspect-ratio: 16/4){
    .back-video{
        width:100%;
        height:auto;
    }
}
@media (max-aspect-ratio: 16/4){
    .back-video{
        width:auto;
        height:100%;
    }
}
</style>
</head>

```

```

<!-- Latest compiled JavaScript -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<nav class="navbar">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">
      <div class="nav-brand d-flex">
        
        <h3>E-MARKET FOR FARMERS</h3>
      </div>
    </a>
    <div class="dropdown">
      <button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton2"
data-bs-toggle="dropdown" aria-expanded="false">
        Language
      </button>
      <ul class="dropdown-menu dropdown-menu-light" aria-labelledby="dropdownMenuButton2">
        <li><a class="dropdown-item active" href="#">English</a></li>
        <li><a class="dropdown-item" href="#">Telugu</a></li>
      </ul>
    </div>
  </div>
</nav>

</div>
<section id="actions">
  <video autoplay loop muted plays-inline class="back-video">
    <source src="{ % static 'img/farm.mp4' % }" type="video/mp4">
  </video>
  <div class="container">
    <div class="row">
      <div class="card" >
        
        <a href="{ % url 'about' % }" class="btn btn-primary">About</a>
      </div>

      <div class="card" >
        
        <a href="{ % url 'myads' % }" class="btn btn-primary">My adds</a>
      </div>

      <div class="card" >
        
        <a href="{ % url 'newad' % }" class="btn btn-primary">Create New ADD</a>
      </div>
    </div>
  </div>

```

```

        <div class="card" >
            
            <a href="notifications.html" class="btn btn-primary">Notification</a>
        </div>

        <div class="card" >
            
            <a href="testimonials.html" class="btn btn-primary">Testimonials</a>
        </div>
    </div>
</div>
</section>

</body>
</html>

```

HOME.CSS

```

*{
    margin:0%;
    padding:0%;
}
body,html {
    overflow-x: hidden;
    font-family: sans-serif;
    font-size: 16px;
}
.back-video
{
    position:absolute;

    width: 100%;
    /* right:0; */
    /* bottom:1; */
    /* z-index: index -1; */

}
.navbar
{
    background-color: rgb(158, 247, 192);
    display:flex;
    font-family: Georgia, 'Times New Roman', Times, serif;
}

h3{
    padding:10px;
}

```

```

#actions {
  background-
image:url("https://dm0qx8t0i9gc9.cloudfront.net/watermarks/video/V7hIImKeijyy0rxl/videoblocks-
beautiful-farm-landscape-rural-aerial-drone-shot-at-
sunset_sbzs6zey8h__de2eb75ec81aac67347abb3b1d9e3459__P360.mp4");
  background-size: cover;
  padding: 0px 0;
}
.container
{
  width:80%;
  margin: auto;

  /* padding: 30px; */
}

.row {
  display: flex;
  flex-wrap: wrap;
  gap: 30px;
  align-items: center;
  justify-content: center;
}
.card {
  width: 300px;
  padding: 0;
  margin-top: 30px;
}

.card img {
  max-width: 100%;
  height: 200px;
}

@media (min-aspect-ratio: 16/4){
  .back-video{
    width:100%;
    height:auto;
  }
}

@media (max-aspect-ratio: 16/4){
  .back-video{
    width:auto;
    height:100%;
  }
}

```

CREATE.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>create add</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
crossorigin="anonymous">

  <link rel="stylesheet" href="create.css">
</head>
<body>

</body>
</html><!-- Latest compiled JavaScript -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
<nav class="navbar ">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">
      <div class="nav-brand d-flex ">
        
        <h3> E-MARKET FOR FARMERS</h3>
      </div>
    </a>
    <div class="dropdown">
      <button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton2"
data-bs-toggle="dropdown" aria-expanded="false">
        Language
      </button>
      <ul class="dropdown-menu dropdown-menu-light" aria-labelledby="dropdownMenuButton2">
        <li><a class="dropdown-item active" href="#">English</a></li>
        <li><a class="dropdown-item" href="#">Telugu</a></li>
      </ul>
    </div>
  </div>
</nav>
</div>
<div class="cont">
  <!-- <marquee behaviour="scroll" direction="left"><h1 style="color:rgb(229, 43, 105);padding-top:
70px; font-size: 40px;">seletct the category</h1></marquee> -->
```



```

    <marquee><h1 style="padding-top:40px;"><span class="rainbow">Select the
category</span></h1></marquee>

</div>

<div class="container">
  <div class="row">
    <div class="card" >
      
      <a href="commodities.html" class="btn btn-primary">commodities</a>
    </div>

    <div class="card" >
      
      <a href="cattle.html" class="btn btn-primary">cattle</a>
    </div>

    <div class="card" >
      
      <a href="village.html" class="btn btn-primary">Village products</a>
    </div>
  </div>
</div>

</body>
</html>

```

```

</body>

```

CREATE.CSS

```

*{
  margin:0%;
  padding:0%;
}
.rainbow {
  background-image: -webkit-gradient( linear, left top, right top, color-stop(0, #f22), color-stop(0.15, #f2f), color-stop(0.3, #22f), color-stop(0.45, #2ff), color-stop(0.6, #2f2), color-stop(1, #f22) );
  background-image: gradient( linear, left top, right top, color-stop(0, #f22), color-stop(0.15, #f2f), color-stop(0.3, #22f), color-stop(0.45, #2ff), color-stop(0.75, #2f2), color-stop(0.9, #ff2), color-stop(1, #f22) );
  color:transparent;
  -webkit-background-clip: text;
  background-clip: text;
}
body,html {
  overflow-x: hidden;
  font-family: sans-serif;
  font-size: 16px;
}

```

```

}
.back-video
{
  position: absolute;

  width: 100%;
  /* right: 0; */
  /* bottom: 1; */
  /* z-index: index -1; */

}

.navbar
{
  background-color: rgb(158, 247, 192);
  display: flex;
  font-family: Georgia, 'Times New Roman', Times, serif;
}

h3{
  padding: 10px;
}

#actions {
  background-
image: url("https://dm0qx8t0i9gc9.cloudfront.net/watermarks/video/V7hI1imKeijyy0rxl/videoblocks-
beautiful-farm-landscape-rural-aerial-drone-shot-at-
sunset_sbzs6zey8h__de2eb75ec81aac67347abb3b1d9e3459__P360.mp4");
  background-size: cover;
  padding: 0px 0;
}

.container
{
  width: 80%;
  margin: auto;

  /* padding: 30px; */
}

.row {
  display: flex;
  flex-wrap: wrap;
  gap: 30px;
  align-items: center;
  justify-content: center;
}

.card {
  width: 300px;
  padding: 0;

```

```

    margin-top: 30px;
}

.card img {
    max-width: 100%;
    height: 200px;
}

@media (min-aspect-ratio: 16/4){
    .back-video{
        width:100%;
        height:auto;
    }
}

@media (max-aspect-ratio: 16/4){
    .back-video{
        width:auto;
        height:100%;
    }
}

```

ABOUT.HTML

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>About</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">

<!-- Latest compiled JavaScript -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>

<style>
    *{
margin:0%;
padding:0%;
    }
    body,html {
        overflow-x: hidden;
        font-family: sans-serif;
        font-size: 16px;
    }
    .back-video
    {
        position:absolute;

```

```

width: 100%;
/* right:0; */
/* bottom:1; */
/* z-index: index -1; */

}
.navbar
{
  background-color: rgb(158, 247, 192);
  display:flex;
  font-family: Georgia, 'Times New Roman', Times, serif;
}

h3{
  padding:10px;
}
#actions {
  background-
image:url("https://dm0qx8t0i9gc9.cloudfront.net/watermarks/video/V7hI1imKeijyy0rx1/videoblocks-
beautiful-farm-landscape-rural-aerial-drone-shot-at-
sunset_sbzs6zey8h__de2eb75ec81aac67347abb3b1d9e3459__P360.mp4");
  background-size: cover;
  padding: 0px 0;
}
.container
{
  width:80%;
  margin: auto;

  /* padding: 30px; */
}

.row {
  display: flex;
  flex-wrap: wrap;
  gap: 30px;
  align-items: center;
  justify-content: center;
}
.card {
  width: 300px;
  padding: 0;
  margin-top: 30px;
}

.card img {

```

```

max-width: 100%;
height: 200px;

}
@media (min-aspect-ratio: 16/4){
.back-video{
width:100%;
height:auto;
}
}
@media (max-aspect-ratio: 16/4){
.back-video{
width:auto;
height:100%;
}
}
</style>
</head>
<body>
<nav class="navbar ">
<div class="container-fluid">
<a class="navbar-brand" href="#">
<div class="nav-brand d-flex ">

<h3>E-MARKET FOR FARMERS</h3>
</div>
</a>
<div class="dropdown">
<button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton2"
data-bs-toggle="dropdown" aria-expanded="false">
Language
</button>
<ul class="dropdown-menu dropdown-menu-light" aria-labelledby="dropdownMenuButton2">
<li><a class="dropdown-item active" href="#">English</a></li>
<li><a class="dropdown-item" href="#">Telugu</a></li>
</ul>
</div>
</div>
</nav>
<div class="container1">
<center></center>
<h6 style="padding: 0px 30px;text-align:justify;font-size:15px;line-height: 1.5em;"> farmer is a

```

person engaged in agriculture, raising living organisms for food or raw materials.

The term usually applies to people who do some combination of raising field crops, orchards, vineyards, poultry, or other livestock. A farmer might own the farmed land or might work as a laborer on land owned by others, but in most developed economies, a farmer is usually a farm owner,

while employees of the farm are known as farm workers, or farmhands. However, in other older definitions

a farmer was a person who promotes or improves the growth of plants, land or crops or raises animals

(as livestock or fish) by labor and attention. Over half a billion farmers are smallholders, most of whom are in developing countries, and who economically support almost two billion people.

Globally, women constitute more than 40% of agricultural employees.

The current concept of a farmers' market is similar to past concepts, but different in

relation to other forms as aspects of consumer retailing, overall, continue to shift over time.

Similar forms existed before the Industrial Age, but often formed part of broader markets, where suppliers

of food and other goods gathered to retail their wares. Trading posts began in 1930s, a shift toward retailers who sold others'

products more than their own. General stores and grocery stores continued that specialization trend in retailing, optimizing the consumer experience,

while abstracting it further from production and from production's growing complexities.

Modern industrial food production's advantages over prior methods depend largely on modern, cheap, fast transport and limited product variability.

But transport costs and delays cannot be eliminated. So where distance strained industrial suppliers' reach, where consumers had strong preference for local variety,

farmers' markets remained competitive with other forms of food retail. Starting in the mid-2000s, consumer demand for foods that are fresher (spend less time in transit)

and for foods with more variety—has led to growth of farmers' markets as a food-retailing mechanism.

<u>Benefits :

To farmers

E-market for farmers can offer farmers increased profit over selling to wholesalers, food processors, or large grocery firms.

By selling directly to consumers, produce often needs less transport, less handling, less refrigeration and less time in storage.

By selling in an outdoor market, the cost of land, buildings, lighting and air-conditioning is also reduced or eliminated.

Farmers may also retain profit on produce not sold to consumers, by selling the excess to canneries and other food-processing firms. At the market, farmers can retain the full premium for part of their produce, instead of only a processor's wholesale price for the entire lot.

However, other economists say "there are relatively few benefits in terms of energy efficiency, quality or cost ... fun though they are, are not good economic models.

Some farmers prefer the simplicity, immediacy, transparency and independence of selling direct to consumers.

One method noted by the special interest group Food Empowerment Project promotes community-supported agriculture programs (CSAs).

In this scheme, consumers pay farms seasonally or monthly to receive weekly or biweekly boxes of produce. Alternatively, they may be required to pay

for an entire season's worth of produce in advance of the growing season. In either case, consumers risk losing their money if there is a crop failure.</h6>

<h4 style="padding:0px 30px;text-align:justify;line-height: 1.5em;" >To Consumers</h4>

<h6 style="padding:0px 30px;text-align:justify;line-height: 1.5em;">To consumers

Some consumers may favor E-market for Farmers for the perceived:

Reduced overhead: driving, parking, etc.

Fresher foods

Seasonal foods

 Healthier foods

A better variety of foods, e.g.: organic foods, pasture-raised meats, free-range eggs and poultry, handmade farmstead cheeses, heirloom produce heritage breeds of meat and many less transport-immune cultivars disfavored by large grocers.

A place to meet neighbors, chat, etc.

A place to enjoy an outdoor walk while getting needed groceries.

Evidence seems to show that overall prices at a typical farmers' market are lower than prices at a supermarket because the process of production is more concise; there is less distance to travel and fewer middlemen. </h6>

</div>

</body>

</html>

COMMODITIES.HTML

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"

integrity="sha384-

EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspD3yD65VohhpUuCOMLASjC"

crossorigin="anonymous">

<link rel="stylesheet" href="commodities.css">

<title>commodities</title>

</head>

<!-- Latest compiled JavaScript -->

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>

</head>

<body>

<nav class="navbar ">

<div class="container-fluid">

<div class="nav-brand d-flex ">

```

    
    <h3> E-MARKET FOR FARMERS</h3>
  </div>
</a>
<div class="dropdown">
  <button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton2"
data-bs-toggle="dropdown" aria-expanded="false">
    Language
  </button>
  <ul class="dropdown-menu dropdown-menu-light" aria-labelledby="dropdownMenuButton2">
    <li><a class="dropdown-item active" href="#">English</a></li>
    <li><a class="dropdown-item" href="#">Telugu</a></li>
  </ul>
</div>
</div>
</nav>
</div>
<div class="cont">
  <!-- <marquee behaviour="scroll" direction="left"><h1 style="color:rgb(229, 43, 105);padding-top:
70px; font-size: 40px;">seletct the category</h1></marquee> -->
  <marquee><h1 style="padding-top:40px;"><span class="rainbow">Select the sub
category</span></h1></marquee>

</div>
<div class="container">
  <div class="row">
    <div class="card" >
      
      <a href="new post.html" class="btn btn-primary">paddy</a>
    </div>

    <div class="card" >
      
      <a href="new post.html" class="btn btn-primary">cotton</a>
    </div>

    <div class="card" >
      
      <a href="new post.html" class="btn btn-primary">turmeric</a>
    </div>

    <div class="card" >
      
      <a href="new post.html" class="btn btn-primary">chillies</a>
    </div>

    <div class="card" >
      

```



```

    <a href="new post.html" class="btn btn-primary">wheat</a>
</div>

<div class="card" >
    
    <a href="new post.html" class="btn btn-primary">sugarcane</a>
</div>
<div class="card" >
    
    <a href="new post.html" class="btn btn-primary">tomato</a>
</div>

<div class="card" >
    
    <a href="new post.html" class="btn btn-primary">onion</a>
</div>

<div class="card" >
    
    <a href="new post.html" class="btn btn-primary">watermelon</a>
</div>
<div class="card" >
    
    <a href="new post.html" class="btn btn-primary">Banana</a>
</div>

<div class="card" >
    
    <a href="new post.html" class="btn btn-primary">Jaggery</a>
</div>

<div class="card" >
    
    <a href="new post.html" class="btn btn-primary">carrot</a>
</div>
<div class="card" >
    
    <a href="new post.html" class="btn btn-primary">Mango</a>
</div>

<div class="card" >
    
    <a href="new post.html" class="btn btn-primary">Brinjal</a>
</div>

<div class="card" >
    
    <a href="new post.html" class="btn btn-primary">Bamboo trees</a>

```

```

        </div>

</div>
</div>

</body>
</html>
COMMODITIES.CSS

*{
    margin:0%;
    padding:0%;
}
body,html {
    overflow-x: hidden;
    font-family: sans-serif;
    font-size: 16px;
}
.rainbow {
    background-image: -webkit-gradient( linear, left top, right top, color-stop(0, #f22), color-stop(0.15,
#f2f), color-stop(0.3, #22f), color-stop(0.45, #2ff), color-stop(0.6, #2f2), color-stop(1, #f22) );
    background-image: gradient( linear, left top, right top, color-stop(0, #f22), color-stop(0.15, #f2f), color-
stop(0.3, #22f), color-stop(0.45, #2ff),color-stop(0.75, #2f2), color-stop(0.9, #ff2), color-stop(1, #f22) );
    color:transparent;
    -webkit-background-clip: text;
    background-clip: text;

}
.navbar
{
    background-color: rgb(158, 247, 192);
    display:flex;
    font-family: Georgia, 'Times New Roman', Times, serif;
}

h3{
    padding:10px;
}
.container
{
    width:80%;
    margin: auto;

    /* padding: 30px; */
}

```

```
.row {
  display: flex;
  flex-wrap: wrap;
  gap: 30px;

  justify-content: center;
}

.card {
  width: 150px;
  padding: 0;
  margin-top: 30px;
}

.card img {
  max-width: 100%;
  height: 100px;
}
```

CATTLE.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIIm9Nao0Yz1ztcQTWfSpd3yD65VohhpUuCOmLASjC"
crossorigin="anonymous">
  <link rel="stylesheet" href="commodities.css">
  <title>cattle</title>
</head>

<!-- Latest compiled JavaScript -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body>
  <nav class="navbar ">
    <div class="container-fluid">
      <a class="navbar-brand" href="#">
        <div class="nav-brand d-flex ">
          
          <h3> E-MARKET FOR FARMERS</h3>
        </div>
      </a>
```

```

<div class="dropdown">
  <button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton2"
data-bs-toggle="dropdown" aria-expanded="false">
    Language
  </button>
  <ul class="dropdown-menu dropdown-menu-light" aria-labelledby="dropdownMenuButton2">
    <li><a class="dropdown-item active" href="#">English</a></li>
    <li><a class="dropdown-item" href="#">Telugu</a></li>
  </ul>
</div>
</div>
</nav>
</div>
<div class="cont">
  <!-- <marquee behaviour="scroll" direction="left"><h1 style="color:rgb(229, 43, 105);padding-top:
70px; font-size: 40px;">seletct the category</h1></marquee> -->
  <marquee><h1 style="padding-top:40px;"><span class="rainbow">Select the sub
category</span></h1></marquee>

</div>
<div class="container">
  <div class="row">
    <div class="card">
      
      <a href="new post.html" class="btn btn-primary">Cow</a>
    </div>

    <div class="card">
      
      <a href="new post.html" class="btn btn-primary">Goat</a>
    </div>

    <div class="card" >
      
      <a href="new post.html" class="btn btn-primary">Sheep</a>
    </div>
    <div class="card">
      
      <a href="new post.html" class="btn btn-primary">Ox</a>
    </div>

    <div class="card">
      
      <a href="new post.html" class="btn btn-primary">Fish</a>
    </div>

    <div class="card">
      

```

```

        <a href="new post.html" class="btn btn-primary">Cock</a>
    </div>
<div class="card">
    
    <a href="new post.html" class="btn btn-primary">Hen</a>
</div>

<div class="card">
    
    <a href="new post.html" class="btn btn-primary">Prawns</a>
</div>

<div class="card">
    
    <a href="new post.html" class="btn btn-primary">Buffallo</a>
</div>
<div class="card">
    
    <a href="new post.html" class="btn btn-primary">Ducks</a>
</div>

<div class="card">
    
    <a href="new post.html" class="btn btn-primary">Dogs</a>
</div>

<div class="card">
    
    <a href="new post.html" class="btn btn-primary">Horse</a>
</div>
<div class="card">
    
    <a href="new post.html" class="btn btn-primary">Turkey Hen</a>
</div>

</div>
</div>

</body>
</html>

```

CREATING AN ADD

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>create add</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOMLASjC"
crossorigin="anonymous">

  <link rel="stylesheet" href="new post.css">

  <style>
    *{
    margin:0%;
    padding:0%;
  }

body,html {
  overflow-x: hidden;
  font-family: sans-serif;
  font-size: 16px;
}

.navbar
{
  background-color: rgb(158, 247, 192);
  display: flex;
  font-family: Georgia, 'Times New Roman', Times, serif;
}

h3{
  padding:10px;
}

.rainbow {
  background-image: -webkit-gradient( linear, left top, right top, color-stop(0, #f22), color-stop(0.15,
#f2f), color-stop(0.3, #22f), color-stop(0.45, #2ff), color-stop(0.6, #2f2), color-stop(1, #f22) );
  background-image: gradient( linear, left top, right top, color-stop(0, #f22), color-stop(0.15, #f2f), color-
stop(0.3, #22f), color-stop(0.45, #2ff),color-stop(0.75, #2f2), color-stop(0.9, #ff2), color-stop(1, #f22) );
  color:transparent;
  -webkit-background-clip: text;
  background-clip: text;
```

```

}
.container
{
    width:70%;
    border-radius: 20px;
    background-color: rgb(219, 227, 219);

}
.form-label
{
    display:flex;
    justify-content:space-between;
    font-size: 15px;
    font-family: Verdana, Geneva, Tahoma, sans-serif;

}

</style>
</head>

</html><!-- Latest compiled JavaScript -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
</head>
<body >
    <nav class="navbar ">
        <div class="container-fluid">
            <a class="navbar-brand" href="#">
                <div class="nav-brand d-flex ">

                    <a href="{ % url 'home' % }"></a>
                    <h3> E-MARKET FOR FARMERS</h3>

                </div>
            </a>
            <div class="dropdown">
                <button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton2"
data-bs-toggle="dropdown" aria-expanded="false">
                    Language
                </button>
                <ul class="dropdown-menu dropdown-menu-light" aria-labelledby="dropdownMenuButton2">
                    <li><a class="dropdown-item active" href="#">English</a></li>
                    <li><a class="dropdown-item" href="#">Telugu</a></li>
                </ul>
            </div>
        </div>
    </nav>

```

```

    </div>
</nav>

<div class="cont">
    <!-- <marquee behaviour="scroll" direction="left"><h1 style="color:rgb(229, 43, 105);padding-top:
70px; font-size: 40px;">seletct the category</h1></marquee> -->
    <center> <h1 style="padding-top:40px;"><span class="rainbow">New post</span></h1></center>
</div>
<div class="container" style="border:rgb(16, 183, 234); border-width:3px; border-style:solid;">
<form action="connect.php" method="post" >

    <label class="slim-field-right" for="text">
    <label for="pick" style="color: rgb(4, 4, 33); font-size:20px;">category :</label>
    <select name="category" id="items">
        <option value="commodities">commodities</option>
        <option value="catle">cattle</option>
        <option value="village products">village products</option>
    </select>
    </label>
<hr>
<label for="pick" style="color: rgb(4, 4, 33); font-size:20px;">Title</label>
<input type="text" id="number" name="date" style="border-radius:5px; padding:6px;" >
<hr>

    <label for="pick" style="color: rgb(4, 4, 33); font-size:20px;">Address :</label><br><br>
<label class="slim-field-right" for="text">
    <span class="form-label">Door no :</span>
    <input id="value" name="locality" value="number" required style="border-radius:5px; padding:4px;"
/>
    </label>

    <label class="slim-field-right" for="text">
    <span class="form-label">village name :</span>
    <input id="text" name="locality" value="village name" required style="border-radius:5px;
padding:4px;"/>
    </label>
    <label class="slim-field-right" for="text">
    <span class="form-label">phone number:</span>
    <input id="text" name="locality" value="phone number" required style="border-radius:5px;
padding:4px;"/>
    </label>
    <label class="slim-field-right" for="text">
    <span class="form-label">mandal :</span>
    <input id="text" name="locality" value="mandal" required style="border-radius:5px; padding:4px;"
/>
    </label>
    <label class="slim-field-right" for="text">

```



```

    <span class="form-label">city :</span>
    <input id="text" name="locality" value="city" required style="border-radius:5px; padding:4px;" />
</label>

<label class="slim-field-right" for="postal_code">
    <span class="form-label">Postal code :</span>
    <input id="number" name="postcode" required style="border-radius:5px; padding:4px;" />
</label>

<button type="button" class="my-button" style="border-radius:5px; padding:8px;background-color:
rgb(112, 248, 97);">Save address </button>
    <input type="reset" value="Clear form" style="border-radius:5px; padding:5px;" />
    <br>
<hr>

    <label for="pick" style="color: rgb(4, 4, 33); font-size:20px;">Available up to :</label>
    <input type="text" id="number" name="date" placeholder="dd/mm/yyyy" style="border-radius:5px;
padding:6px;" >
<hr>

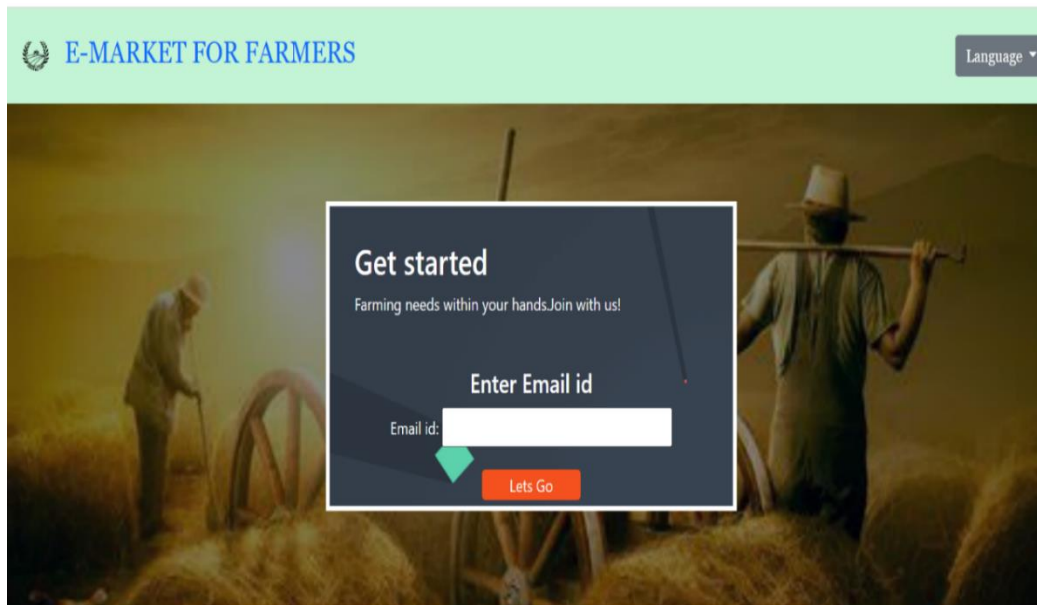
    <label for="pick" style="color: rgb(4, 4, 33); font-size:20px;">Description :</label>
    <textarea name="comments" id="comments" style="width:90%;height:50px;padding:2%;font: size
1em;border:0.5px solid #161f0d;"></textarea><br>
<hr>
<label for="pick" style="color: rgb(4, 4, 33); font-size:20px;">Upload images :</label>
<br>
<form action="connect.php" method="post" enctype="multipart/form-data">
    <label>Select a file:</label>
    <input type="file" id="myfile" name="image"><br><br>
    <input type="submit" name="submit" value="upload">
</form>

</form>
</div>
</body>
</html>

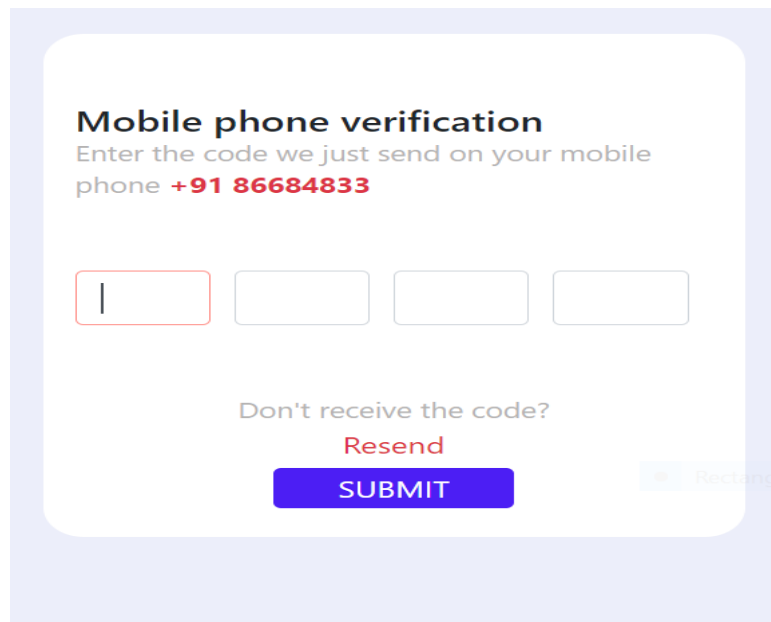
```

8. SCREENSHOTS

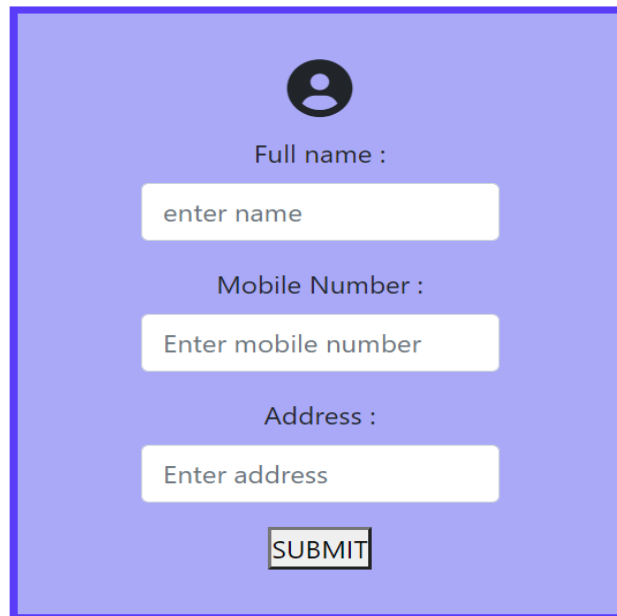
9.1 USER SCREEN



9.2 VERIFICATION PAGE



9.3 PROFILE



A user profile registration form with a light purple background and a dark purple border. At the top is a black circular icon with a white person silhouette. Below it are three text input fields: 'Full name :', 'Mobile Number :', and 'Address :'. Each field has a placeholder text: 'enter name', 'Enter mobile number', and 'Enter address'. At the bottom is a 'SUBMIT' button with a black border and white text.

Full name :

enter name

Mobile Number :

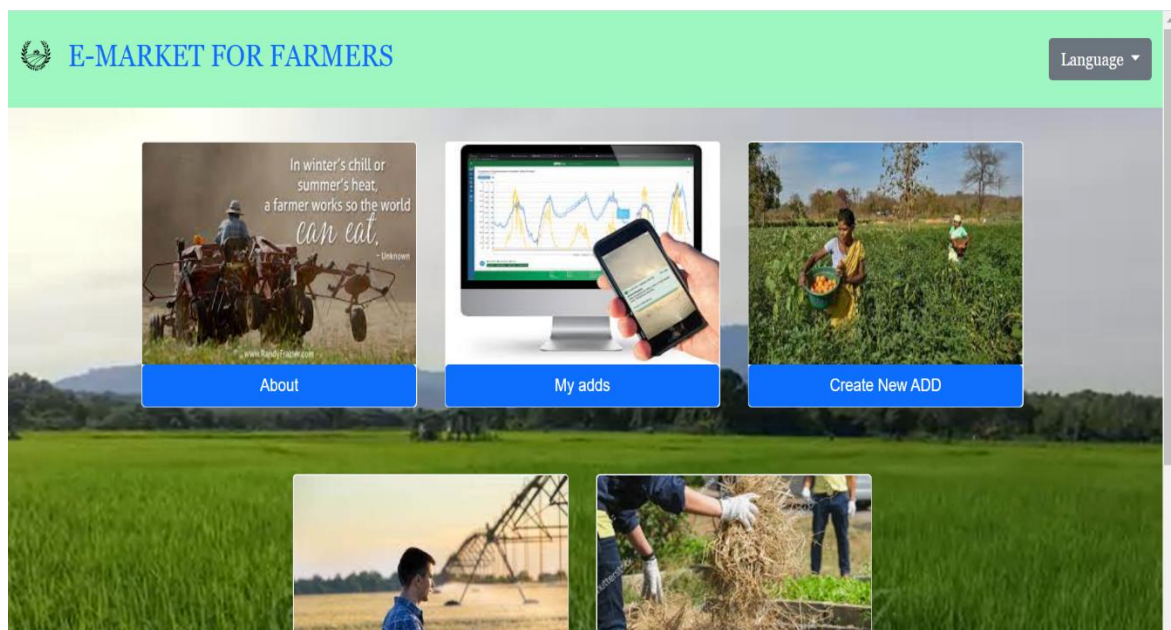
Enter mobile number

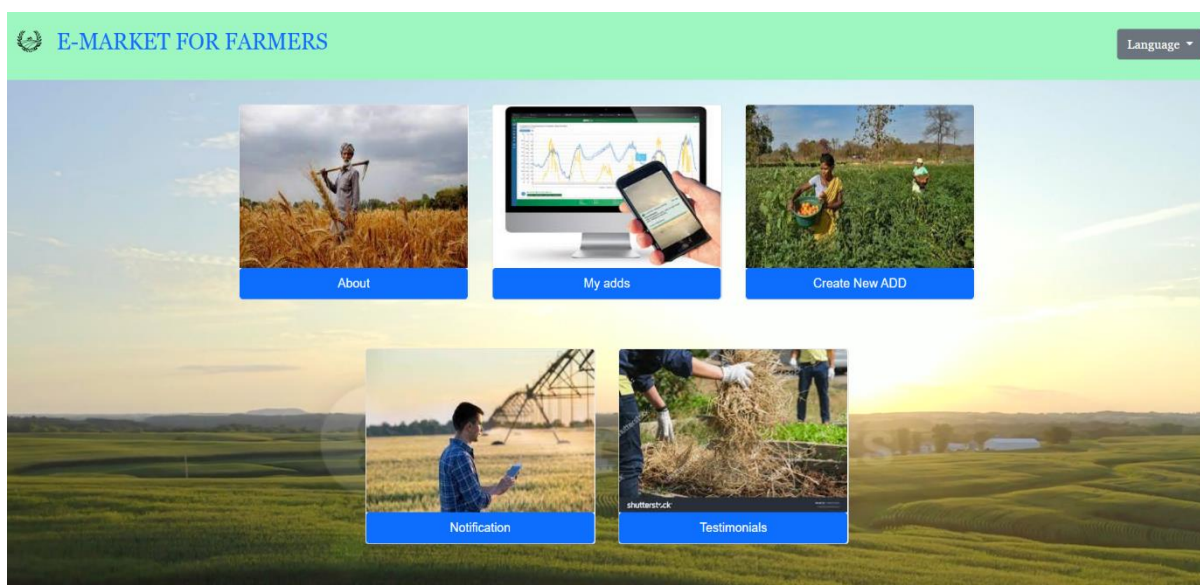
Address :

Enter address

SUBMIT

9.4 HOME PAGE





9.5 ABOUT PAGE:

E-MARKET FOR FARMERS
Language

farmer is a person engaged in agriculture, raising living organisms for food or raw materials. The term usually applies to people who do some combination of raising field crops, orchards, vineyards, poultry, or other livestock. A farmer might own the farmed land or might work as a laborer on land owned by others, but in most developed economies, a farmer is usually a farm owner, while employees of the farm are known as farm workers, or farmhands. However, in other older definitions a farmer was a person who promotes or improves the growth of plants, land or crops or raises animals (as livestock or fish) by labor and attention. Over half a billion farmers are smallholders, most of whom are in developing countries, and who economically support almost two billion people. Globally, women constitute more than 40% of agricultural employees.

The current concept of a farmers' market is similar to past concepts, but different in relation to other forms as aspects of consumer retailing, overall, continue to shift over time. Similar forms existed before the Industrial Age, but often formed part of broader markets, where suppliers of food and other goods gathered to retail their wares. Trading posts began in 1930s, a shift toward retailers who sold others' products more than their own. General stores and grocery stores continued that specialization trend in retailing, optimizing the consumer experience, while abstracting it further from production and from production's growing complexities.

Modern industrial food production's advantages over prior methods depend largely on modern, cheap, fast transport and limited product variability. But transport costs and delays cannot be eliminated. So where distance strained industrial suppliers' reach, where consumers had strong preference for local variety, farmers' markets remained competitive with other forms of food retail. Starting in the mid-2000s, consumer demand for foods that are fresher (spend less time in transit) and for foods with more variety—has led to growth of farmers' markets as a food-retailing mechanism.

Benefits :

To farmers

E-market for farmers can offer farmers increased profit over selling to wholesalers, food processors, or large grocery firms. By selling directly to consumers, produce often needs less transport, less handling, less refrigeration and less time in storage. By selling in an outdoor market, the cost of land, buildings, lighting and air-conditioning is also reduced or eliminated. Farmers may also retain profit on produce not sold to consumers, by selling the excess to canneries and other food-processing firms. At the market, farmers can retain the full premium for part of their produce, instead of only a processor's wholesale price for the entire lot. However, other economists say "There are relatively few benefits in terms of energy efficiency, quality or cost...fun though they are, are not good economic models. Some farmers prefer the simplicity, immediacy, transparency and independence of selling direct to consumers. One method noted by the special interest group Food Empowerment Project promotes community-supported agriculture programs (CSAs). In this scheme, consumers pay farms seasonally or monthly to receive weekly or biweekly boxes of produce. Alternatively, they may be required to pay for an entire season's worth of produce in advance of the growing season. In either case, consumers risk losing their money if there is a crop failure.

To Consumers

To consumers Some consumers may favor E-market for Farmers for the perceived:

- Reduced overhead: driving, parking, etc.
- Fresher foods
- Seasonal foods
- Healthier foods
- A better variety of foods, e.g.: organic foods, pasture-raised meats, free-range eggs and poultry, handmade farmstead cheeses, heirloom produce heritage breeds of meat and many less transport-immune cultivars disavored by large grocers.
- A place to meet neighbors, chat, etc.
- A place to enjoy an outdoor walk while getting needed groceries.

Evidence seems to show that overall prices at a typical farmers' market are lower than prices at a supermarket because the process of production is more concise, there is less distance to travel and fewer middlemen.

9.6 MY AD PAGE

My Ads



Rice

iasdhfasdk hfasd f

Phone : 392832832382

Available upto : May 4, 2022



Jersey Cow Maharastra

The Jersey is a British breed of small dairy cattle from Jersey, in the British Channel Islands. It is one of three Channel Island cattle breeds, ...

Height: Female: 115–120 cm (45–47 in) Use: dairy; draught Distribution:

world-wide Coat: variable ಇಂಗ್ಲಿಷ್

ನುಂದೆ ಅನುವದించಬಡಿంది-ಪೆರೈ

ನುಂದೆ ಅನುವದించಬಡಿంది-ಪೆರೈ



paddy

I grow paddy . Being an organic farmer, I always look for natural solutions, such as mixing #neem seeds in cow urine to create a natural pesticide. I am all for technology in farming as long as it is ecofriendly. I am happy to see initiatives like FarmDost and many new farmers take

9.7 CREATING AN AD PAGE

E-MARKET FOR FARMERS

Select the category



commodities



cattle



Village products

9.8 COMMODITIES PAGE

Select the sub category



paddy



cotton



turmeric



chillies



wheat



sugarcane



tomato



onion



watermelon



Banana

9.9 CATTLES PAGE



E-MARKET FOR FARMERS

Language ▾

Select the sub category



Cow



Goat



Sheep



Ox



Fish



Cock



Hen



Prawns




Buffalo





Ducks


9.10 VILLAGE PRODUCTS PAGE


 E-MARKET FOR FARMERS Language ▾


Select the sub category



Pickles



flour dishes



Toys



Pots



Liquid foods


coconuts


Sarees


Palm products


Bamboo works


Carpenter works

9.11 POST AN AD PAGE

New post

Post for: ☐ Buy products ☐ Sell products

Type of variety :

Address :

Door no : village name : phone number: mandal : city : Postal code :

number village name phone number mandal city

Quantity(No of Bags):

Expecting Price per Bag/Basket(Rs):

Experting product on or before(date):

Description :

Upload images :

Select a file: No file chosen

10.CONCLUSION:

The main aim of our project is to built a bridge of communication between the farmers and customers across the country so that they can get communicate together and can talk about any product related queries for both ends. The main task will be a challenge to the most of the farmers because they are the lacking the knowledge about the new technology and trends which is used in this fast developing world. The main success of our project is to provide the fruitful benefits for both the customer as well as to the farmers.