# Introduction

ServiceNow Track

1. ServiceNow Fundamentals

- What is ServiceNow?

- Customer service agent they will likely tell you that ServiceNow is a ticketing tool providing the ability to record and resolve customer requests (Ticketing Tools)

- IT platform support team might define ServiceNow a Configuration management tool that keeps track of hardware and software (CMDB)

- If you ask from the development team they would say that ServiceNow is a cloud-based development environment (Development Platform)

- Enterprise Database

- Request Catalog

- Knowledge Management platform

1. ServiceNow Fundamentals

▶ What is ServiceNow?

▶ ServiceNow can mean many different things to many different people with many different perspectives (Personas)
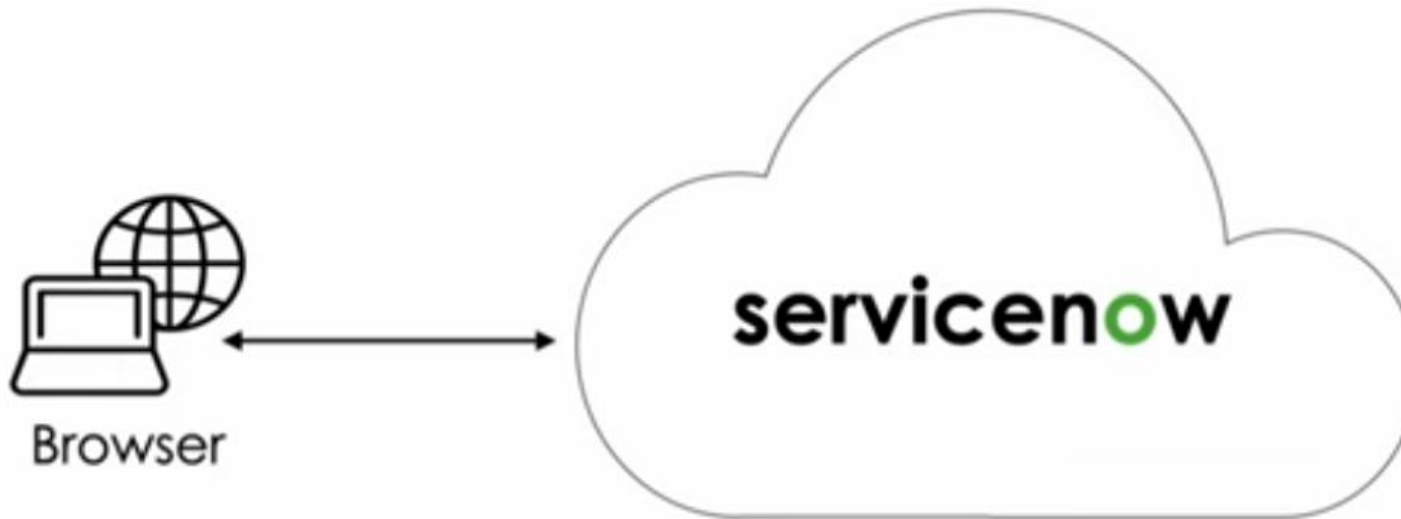
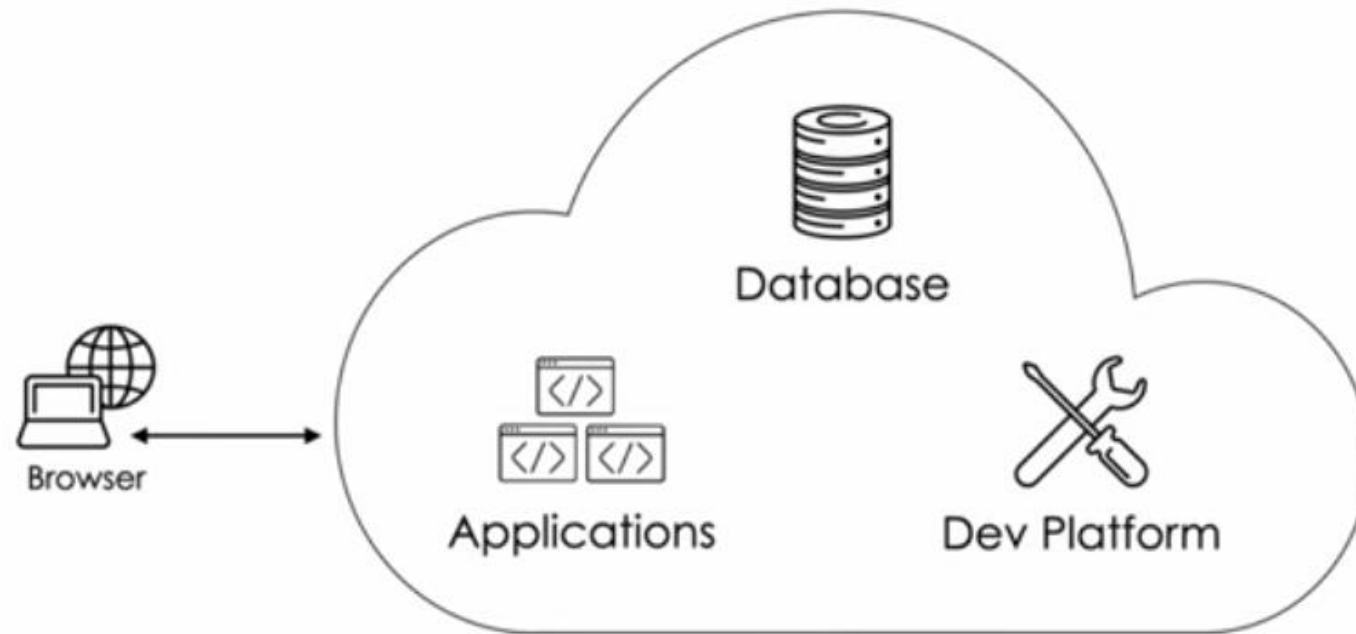# 1. ServiceNow Fundamentals

▶ What is ServiceNow?

▶ ServiceNow can mean many different things to many different people with many different perspectives (Personas)

1. ServiceNow Fundamentals

▶ What is ServiceNow?

▶ ServiceNow is a cloud-based technology platform built to deliver common IT services, based upon best practices and lessons learned.
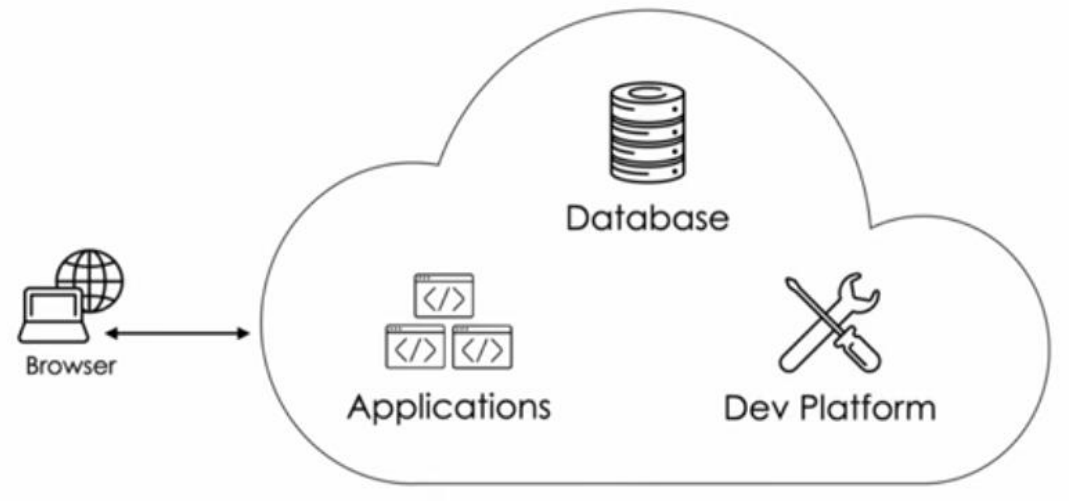
1. ServiceNow Fundamentals

- What is ServiceNow?
- Database:
  - Just about everything in ServiceNow is a record in a database table
  - Single, enterprise-wide (there is only one database)
  - Over 4,0000 tables designed to meet the needs of most common IT services
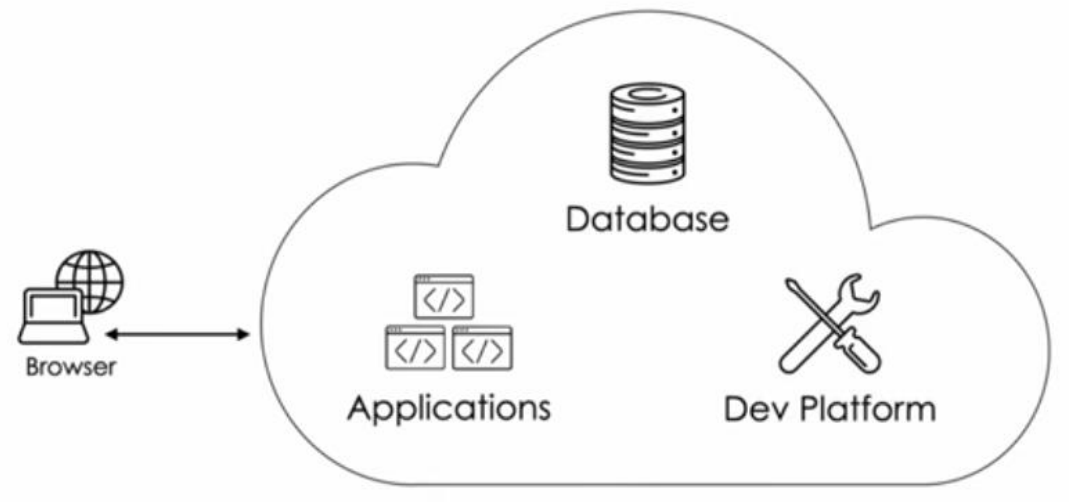
1. ServiceNow Fundamentals

- What is ServiceNow?
- Applications:
    - Hundreds of applications installed out-of-box
    - Business applications to provide services for things like ALM, APM,CM,SecOps,ITSM,etc.
    - ServiceNow Store

1. ServiceNow Fundamentals

- What is ServiceNow?
- Dev Platform:
  - Modify existing (out-of-box) applications or build custom applications
  - No Code / Low Code / Full Code
  - Create custom database tables , workflows, and scripts
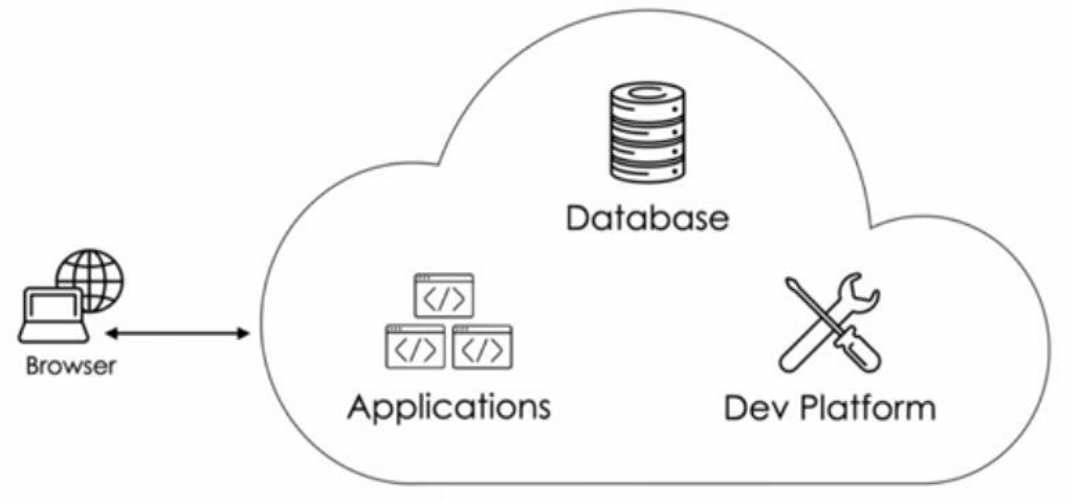
HTML

HTML is the standard markup language for creating Web pages.

1. Modern Web Architecture

- Modern web architecture focuses on **performance, scalability, and user experience**. Typical flow:

- **Client (Browser)**
  Renders UI using HTML, CSS, and JavaScript.

- **Server**
  Handles business logic, authentication, APIs.

- **Data Layer**
  Databases, caches, search engines.

- **Delivery Layer**
 CDNs, edge networks for fast global access.

1. Modern Web Architecture

Common patterns:

▶ **SPA / MPA** (Single vs Multi-Page Apps)

▶ **SSR / CSR / SSG** (Server-Side, Client-Side, Static Generation)

▶ **API-driven** (REST, GraphQL)

▶ **Microservices & Edge rendering**

▶ Goal: **fast load, smooth interaction, and maintainable code.**

1. Modern Web Architecture

## 2. How Browsers Render HTML & CSS (Critical Rendering Path)

▶ The **Critical Rendering Path (CRP)** is the sequence of steps a browser takes to turn code into pixels.
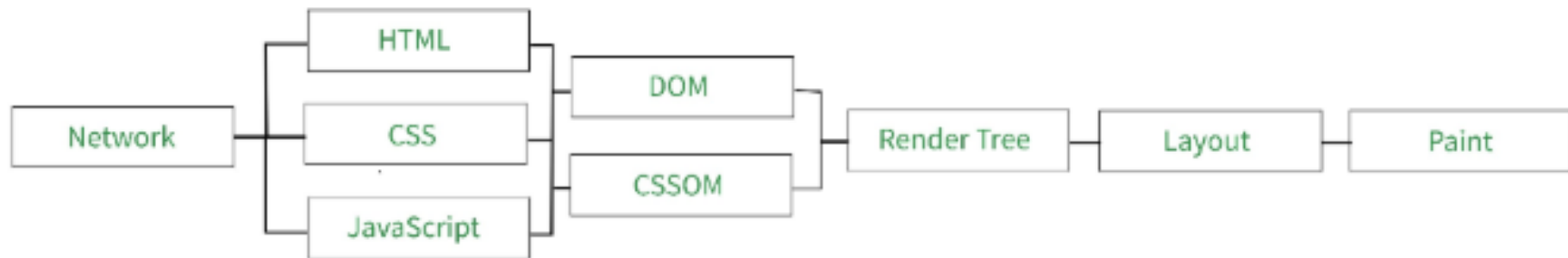
1. Modern Web Architecture

Steps:

- **HTML is parsed → DOM**
  - Browser converts HTML into the **Document Object Model**.
  - DOM is a crucial part of any webpage. HTML is the first file that will be sent from the server when a webpage is loaded. Once the browser starts getting these packets(which contain bytes of code), it will start the parsing instead of waiting for all packets to be received. The browser follows an incremental DOM update approach. In parsing, first, it converts bytes to characters & then into tokens & then into nodes & then finally into the DOM tree. This cycle repeats whenever a new packet is received
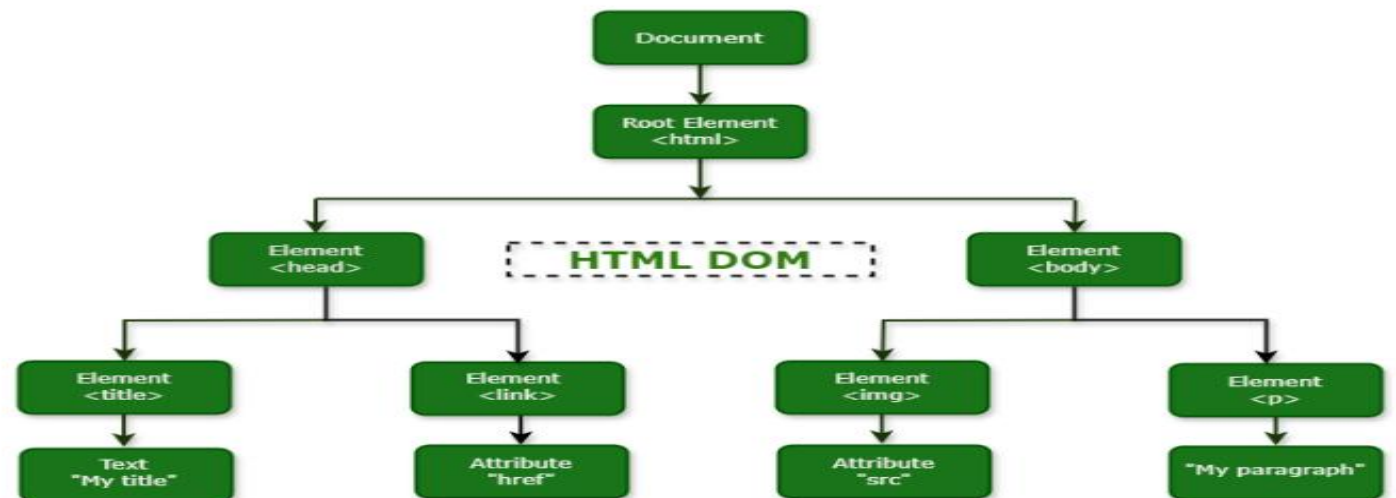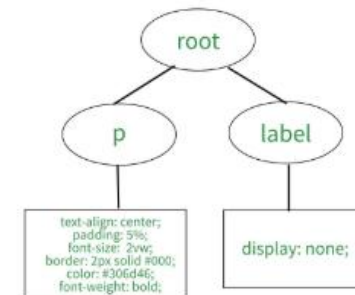
1. Modern Web Architecture

Steps:

▶ **CSS is parsed → CSSOM**

  ▶ All CSS rules are processed into the **CSS Object Model**.

  ▶ **Cascading Style Sheet Object Model (CSSOM):** This object model focuses on the style part of a webpage. CSSOM tree construction is render-blocking, unlike DOM tree construction. Here browser will be waiting for all the packets to be received & starts parsing once all the packets of the file are received because the style properties can be replaced or overwritten in upcoming lines. The browser follows the same approach of converting Bytes to CSSOM tree

Bytes to CSSOM Conversion Flow fig 1.4

1. Modern Web Architecture

Steps:

- **Render Tree**

  - DOM + CSSOM combined (only visible elements).

  - **Render Tree:** Render tree combines both the DOM & CSSOM tree into one. It takes each node from the DOM tree & maps the respective CSS object to it. Each node in the render tree has style properties assigned to it if there is any style content defined for a particular node. One thing to notice in the render tree construction is, it only catches the visible content & adds it to the tree. Since the head segment doesn't have any visibility to it, it won't be there in the render tree.
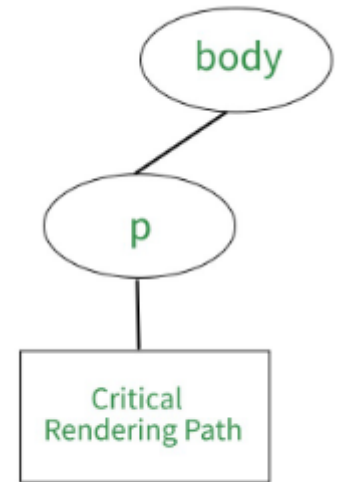
1. Modern Web Architecture

Steps:

- **Layout (Reflow)**
  - Calculates element sizes and positions.
  - **Layout(Visual Formatting Model):**After building the render tree, the layout comes into the picture. The layout concentrates on the placement of elements on the screen based on the size of the screen. The element having a height and width of 100% captures the whole screen. The execution of the layout depends on the nodes of the DOM tree. The more the number, the more time browser takes to build the layout. It works on the algorithm which calculates and uses a bunch of stuff like the box model, floats, and positioning. In order to actually render the page, the browser uses something called *Visual Formatting Model* .

- **Paint**
  - Draws pixels to the screen.

- **Compositing**
  - Layers combined (GPU-accelerated).

1. Modern Web Architecture

Steps:

▶ **Paint**

    ▶ Draws pixels to the screen.

    ▶ This is the final step, after having render-tree & layout, the browser paints the pixels on the screen.

▶ **Compositing**

    ▶ Layers combined (GPU-accelerated).

Critical Rendering Path

1. Modern Web Architecture

Performance tips:

▶ Minimize **render-blocking CSS**

▶ Avoid heavy **layout thrashing**

▶ Use **async/defer** for JS

▶ Reduce **critical CSS size**

1. Modern Web Architecture

3. DOM Overview

▶ The **DOM (Document Object Model)** is a **tree representation** of the HTML document.

▶ Key points:

▶ Each HTML element = **node**

▶ JavaScript interacts with the page via the DOM

▶ DOM is **mutable** (can be changed dynamically)

1. Modern Web Architecture

3. DOM Overview

▶ **Example:**

→ becomes a tree structure in memory.

Important concepts:

- DOM traversal

- Event bubbling & capturing

- DOM manipulation performance (reflows & repaints)

```
html


<div>

  <p>Hello</p>

</div>
```

1. Modern Web Architecture

4. SEO & Accessibility Importance

▶ **SEO (Search Engine Optimization)**

▶ Ensures content is **discoverable by search engines.**

▶ Key factors:

　▶ Semantic HTML (<header>, <article>, <nav>)

　▶ Page speed & Core Web Vitals

　▶ Mobile-friendly design

　▶ Proper metadata (title, description, headings)

　▶ Good SEO = **better visibility & traffic.**

1. Modern Web Architecture

Accessibility (a11y)

▶ Ensures the web is usable by **everyone**, including users with disabilities.

▶ Key practices:

    ▶ Semantic HTML

    ▶ ARIA roles (only when needed)

    ▶ Keyboard navigation

    ▶ Screen-reader support

    ▶ Color contrast & readable fonts

▶ Accessibility benefits:

    ▶ Legal compliance

    ▶ Better UX for all users

    ▶ Improved SEO (search engines "read" like screen readers)

2. Advanced HTML5

1. Semantic HTML (In Depth)

▶ **Semantic HTML** uses elements that clearly describe their meaning and purpose, not just layout.

▶ **Common Semantic Elements**

  ▶ <header> – Intro content or navigation

  ▶ <nav> – Primary navigation links

  ▶ <main> – Main content (one per page)

  ▶ <section> – Thematic grouping of content

  ▶ <article> – Independent, reusable content

  ▶ <aside> – Related but secondary content

  ▶ <footer> – Footer info for a section/page

  ▶ <figure> / <figcaption> – Media with captions

2. Advanced HTML5

Why it matters:

▶ Improves **readability & maintainability**

▶ Enables **screen readers** to understand structure

▶ Helps **search engines** index content properly

▶ Reduces need for excessive ARIA

▶ **Rule of thumb:**

▶ Use semantic HTML first, ARIA only when necessary.

2. Advanced HTML5

2. Accessibility – ARIA Roles & Attributes

▶ **ARIA (Accessible Rich Internet Applications)** helps make dynamic or custom UI components accessible.

▶ **Common ARIA Roles**

  ▶ role="button"

  ▶ role="navigation"

  ▶ role="dialog"

  ▶ role="alert"

  ▶ role="tabpanel"

▶ **Important ARIA Attributes**

  ▶ aria-label – Accessible name

  ▶ aria-labelledby – References visible text

  ▶ aria-hidden – Hides from screen readers

  ▶ aria-expanded – Toggle state

  ▶ aria-live – Announces dynamic updates

2. Advanced HTML5

▶ **Example**

ARIA Best Practices

- **Do not override native semantics**

- Always keep ARIA state **in sync**

- Never use ARIA as a replacement for semantic elements Keyboard support is mandatory

```html
<button aria-expanded="false" aria-controls="menu">
  Menu
</button>
```

2. Advanced HTML5

- **3. SEO-Friendly HTML Structure**
  - Search engines rely heavily on **document structure**.
- **Best Practices**
  - One <h1> per page (main topic)
  - Logical heading order (h1 → h2 → h3)
  - Use semantic containers (<article>, <section>)
  - Meaningful anchor text (avoid "click here")
  - Image alt attributes for content images

2. Advanced HTML5

▶ **Example**

Result:
- Better crawlability
- Higher rankings
- Improved accessibility

```html
html

<header>
  <h1>Web Development Guide</h1>
</header>


<main>
  <article>
    <h2>HTML Basics</h2>
    <p>...</p>
  </article>
</main>


<footer>
  <p>© 2026</p>
</footer>
```

2. Advanced HTML5

▶ **4. Meta Tags (SEO, Viewport, Social Sharing)**

▶ **Essential SEO Meta Tags**

▶ Meta tags are HTML snippets placed in a page's <head> section that provide structured metadata—such as titles, descriptions, and viewport settings—to search engines and browsers. They do not appear on the page itself but control indexing, responsiveness, and how content is displayed in search results.

Key Types of Meta Tags

- **Meta Description:** Summarizes page content for search engine results pages (SERPs).

- **Title Tag:** Defines the page title, critical for search relevance.

- **Viewport:** Controls layout on mobile browsers.

- **Robots:** Instructs crawlers to index or ignore pages.

- **Charset:** Defines character encoding (e.g., UTF-8). 🔗

- description → affects search snippet

- robots → indexing behavior

```html
<meta charset="UTF-8">
<meta name="description" content="Learn advanced HTML5 concepts">
<meta name="robots" content="index, follow">
```

2. Advanced HTML5

▶ **Viewport (Mobile Optimization)**

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

- Critical for **responsive design** and Core Web Vitals.

- The **viewport meta tag** is a small but crucial piece of HTML code that provides instructions to the browser on how to control the page's dimensions and scaling, which is essential for **responsive web design** and ensuring a good user experience on mobile devices.

2. Advanced HTML5

► **Social Sharing Meta Tags**

► **Open Graph (Facebook, LinkedIn)**

```html
html

<meta property="og:title" content="Advanced HTML5">
<meta property="og:description" content="Semantic HTML, ARIA, and SEO">
<meta property="og:image" content="image.jpg">
<meta property="og:url" content="https://example.com">
```

2. Advanced HTML5

▶ **Twitter Cards**

```html
html

<meta name="twitter:card" content="summary_large_image">

<meta name="twitter:title" content="Advanced HTML5">

<meta name="twitter:description" content="HTML5 best practices">
```

3. Forms & Validations

- **1. HTML5 Form Elements**

- HTML5 introduced **powerful built-in form controls** that reduce JavaScript dependency.

- **Core Form Elements**

  - <form> – Form container

  - <label> – Accessible field labeling

  - <input> – Multiple input types

  - <textarea> – Multi-line text

  - <select> / <option> / <optgroup>

  - <button> – Submit / reset / custom actions

  - <fieldset> / <legend> – Group related fields

3. Forms & Validations

- **Best Practice**

- <label for="email">Email</label>

- <input id="email" name="email" type="email">

  - ✓ Improves usability
    ✓ Required for accessibility
    ✓ Helps screen readers

3. Forms & Validations

- **Input Types (Advanced Overview)**

- HTML5 input types provide **built-in validation, keyboards, and UX improvements.**

- **Common Input Types**

Type	Purpose
email	Valid email format
password	Masked input
number	Numeric input
tel	Phone keypad
url	URL validation
date	Date picker
range	Slider input
file	File upload
color	Color picker
search	Search field

3. Forms & Validations

▶ **Example**

```html
html

<input type="email" required>

<input type="date" min="2024-01-01">

<input type="range" min="0" max="100">

<input type="file" accept="image/*">
```

Benefits

- Automatic validation
- Mobile-optimized keyboards
- Consistent UX across browsers

3. Forms & Validations

- **Native Validation**
  - HTML5 provides **constraint validation API** — no JavaScript required for basic rules.
- **Common Validation Attributes**
  - required
  - min / max
  - minlength / maxlength
  - pattern
  - step
  - readonly / disabled

3. Forms & Validations

- ▶ **Example**

```html
html

<input
  type="password"
  minlength="8"
  pattern="(?=.*\d)(?=.*[A-Z]).*"
  required
>
```

3. Forms & Validations

▶ **Validation States**

    ▶ :valid

    ▶ :invalid

    ▶ :required

    ▶ :optional

Pros of Native Validation

- Fast

- Accessible by default

- Minimal JS

- Consistent with browser behavior

```html
html

<input
  type="email"
  oninvalid="this.setCustomValidity('Enter a valid email')"
  oninput="this.setCustomValidity('')"
>
```

```html
html

<input type="email" required placeholder="Enter a valid email">
```

```css
css

input:invalid {
  border-color: crimson;
  background: #ffe6e6;
}
```

3. Forms & Validations

▶ **4. Accessibility in Forms (Critical)**

▶ Forms are **one of the biggest accessibility failure points.**

▶ **1. Proper Labeling**

  ▶ Every input must have a <label>

  ▶ Use for + id

  ▶ Avoid placeholder-only labels

❌ Bad:

```html
<input placeholder="Email">
```

✅ Good:

```html
<label for="email">Email</label>
<input id="email" type="email">
```

3. Forms & Validations

▶ **2. Grouping Related Fields**

```html
html

<fieldset>

  <legend>Billing Address</legend>

  ...

</fieldset>
```

Helps screen readers understand context.

3. Forms & Validations

- **3. Error Messages (Accessible)**
  - Errors must be **programmatically associated**
  - Use aria-describedby
  - Announce errors with aria-live

```html
<input id="email" aria-describedby="emailError">
<span id="emailError" aria-live="assertive">
  Invalid email
</span>
```

3. Forms & Validations

▶ **4. Keyboard Navigation**

   ▶ All fields reachable via Tab

   ▶ No keyboard traps

   ▶ Visible focus styles

▶ **5. Required Fields**

   ▶ Use required attribute

   ▶ Indicate visually **and** programmatically

   ▶ Avoid only using color to indicate errors

```css
/* ✕ Bad */
:focus {
  outline: none;
}
```

CSS

```css
/* ✅ Good */
:focus-visible {
  outline: 3px solid #2563eb;
  outline-offset: 2px;
}
```

Lab Time
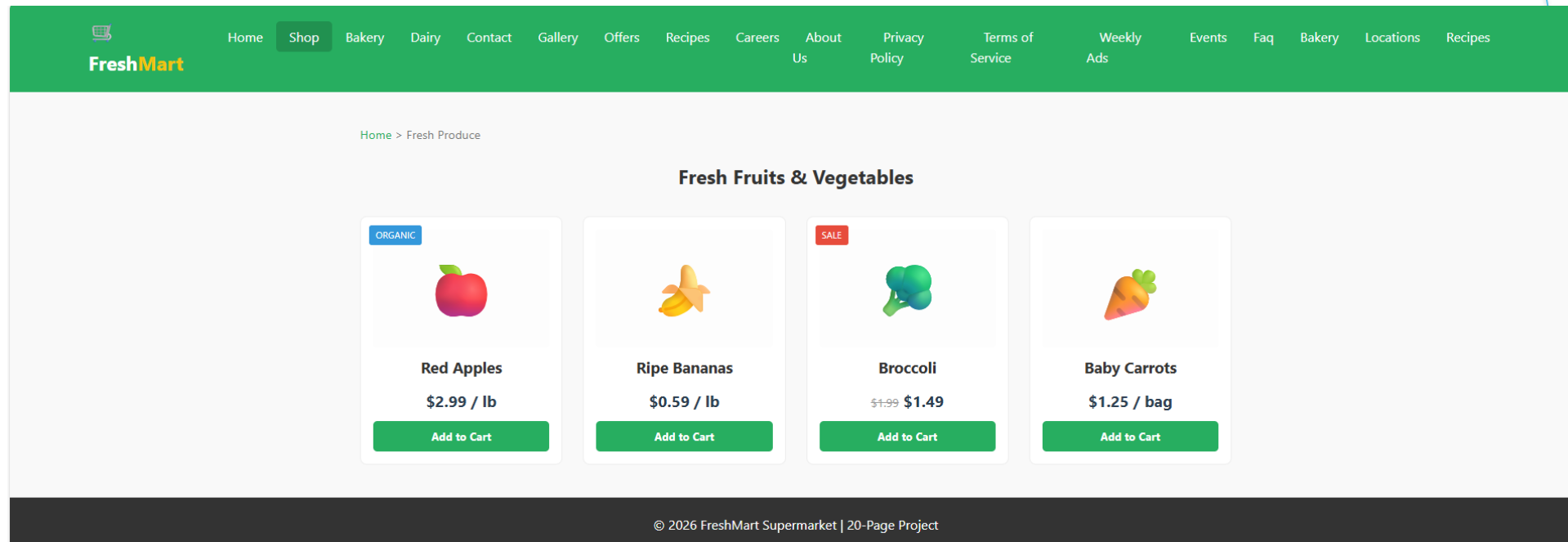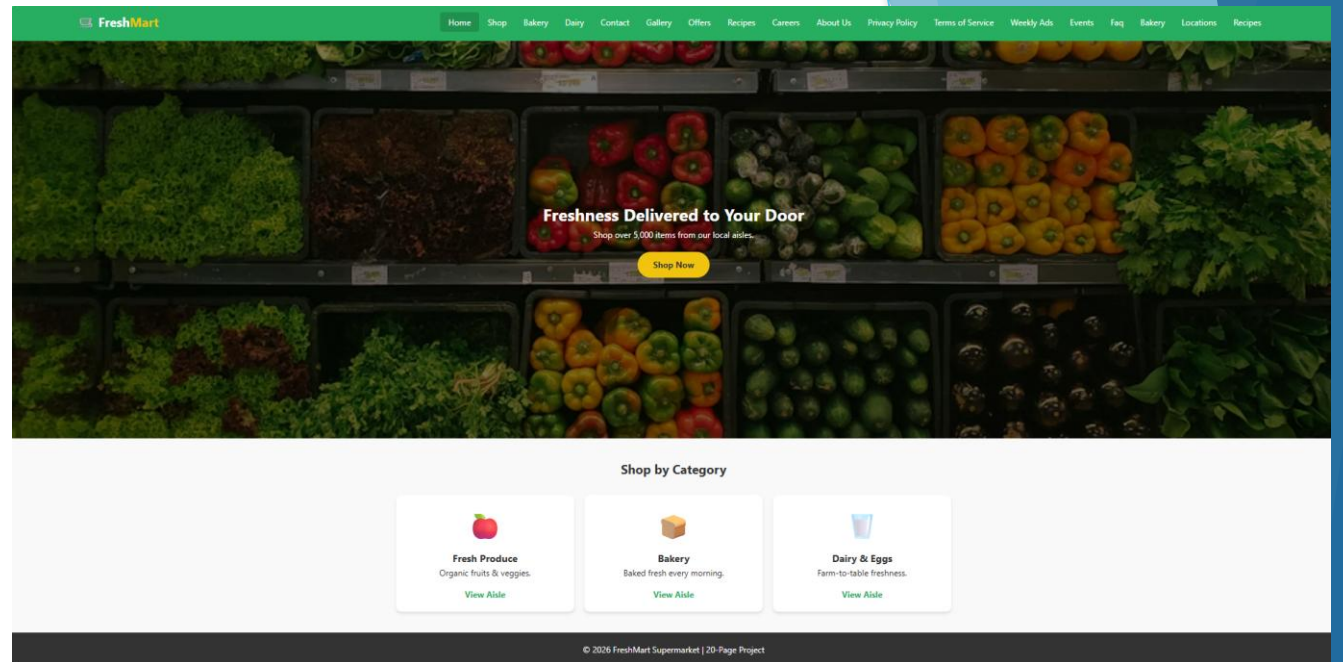
Lab Time

Fresh From Our Oven

French Baguette

$1.50

Add to Cart

Butter Croissant

$2.25

Add to Cart

Blueberry Muffin

$3.00

Add to Cart

Dairy & Cold Fridge

Whole Milk (1 Gallon)

$3.99

Add to Cart

Cheddar Cheese block

$4.50

Add to Cart

Large Brown Eggs (12ct)

$2.80

Add to Cart

Lab Time

Lab Time

🛒
FreshMart

Home Shop Bakery Dairy Contact Gallery Offers Recipes Careers About Us Privacy Policy Terms of Service Weekly Ads Events Faq Bakery Locations Recipes
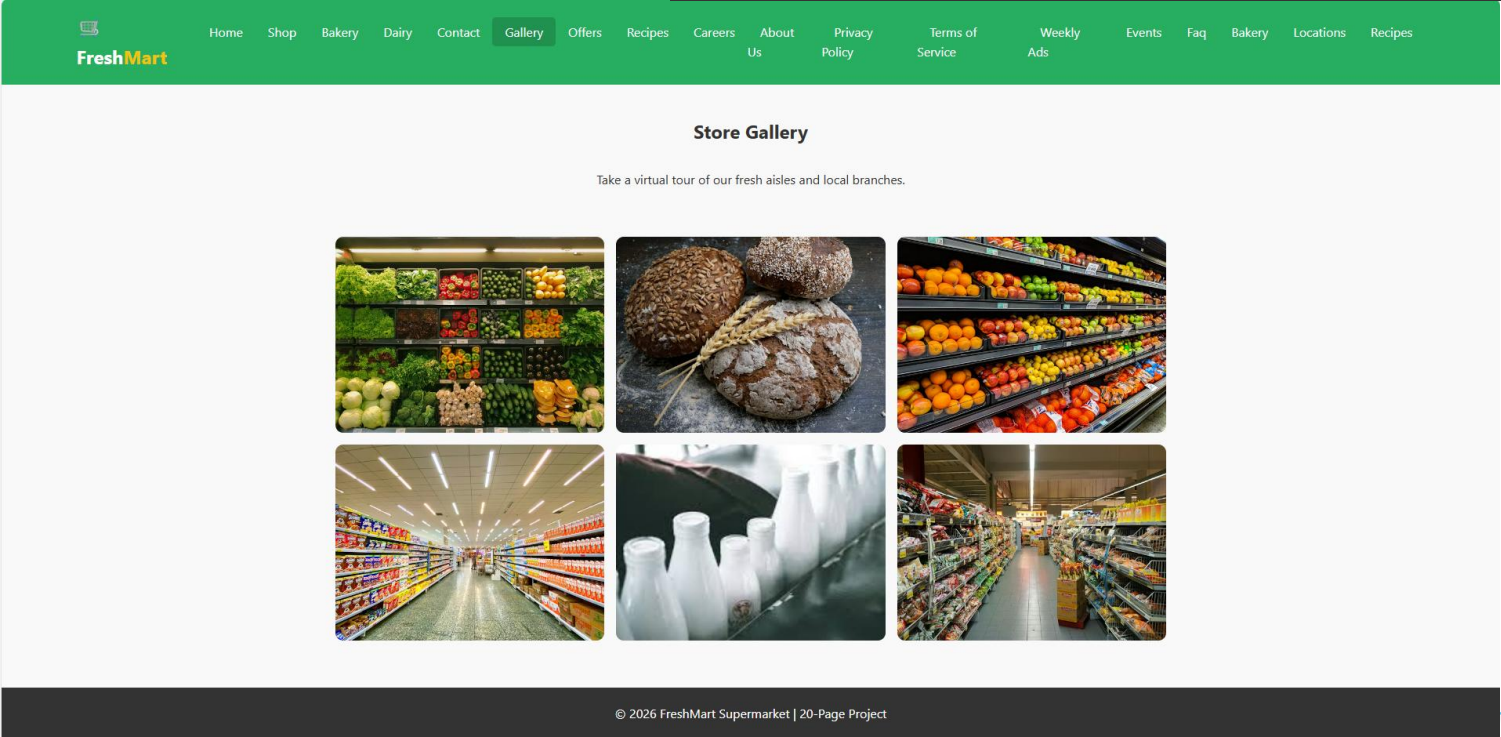
Exclusive Member Rewards

Clip these digital coupons to save big on your next visit!

20% OFF

First Order Discount
Valid on all fresh produce items.

FRESH20

$10 BACK

Weekend Special
Spend $50 or more and save $10.

SAVE10

FREE

Bakery Bonus
Free baguette with any pastry purchase.

YUMMY

🛒
FreshMart

Home Shop Bakery Dairy Contact Gallery Offers Recipes Careers About Us Privacy Policy Terms of Service Weekly Ads Events Faq Bakery Locations Recipes

Cook with FreshMart

Summer Greek Salad
Time: 15 Mins | Difficulty: Easy

View Recipe

Classic Spaghetti
Time: 30 Mins | Difficulty: Medium

View Recipe

Apple Cinnamon Pie
Time: 60 Mins | Difficulty: Hard

View Recipe

Lab Time

🛒 **FreshMart**

Home Shop Bakery Dairy Contact Gallery Offers Recipes Careers About Us Privacy Policy Terms of Service Weekly Ads Events Faq Bakery Locations Recipes

Join the FreshMart Team

Work in a fun, fast-paced environment with great benefits!

Store Cashier
Full-Time | Downtown Branch

Apply Now

Bakery Specialist
Part-Time | Riverside Branch

Apply Now

Delivery Driver
Contract | All Locations

Apply Now

🛒 **FreshMart**

Home Shop Bakery Dairy Contact Gallery Offers Recipes Careers About Us Privacy Policy Terms of Service Weekly Ads Events Faq Bakery Locations Recipes
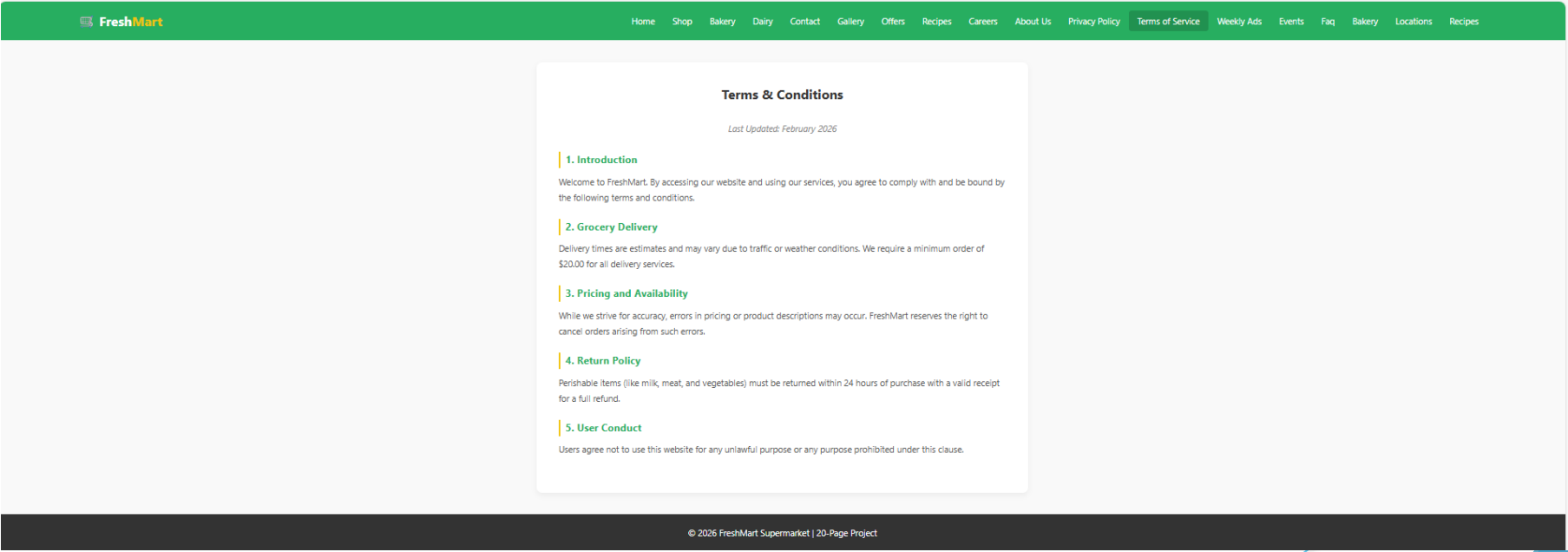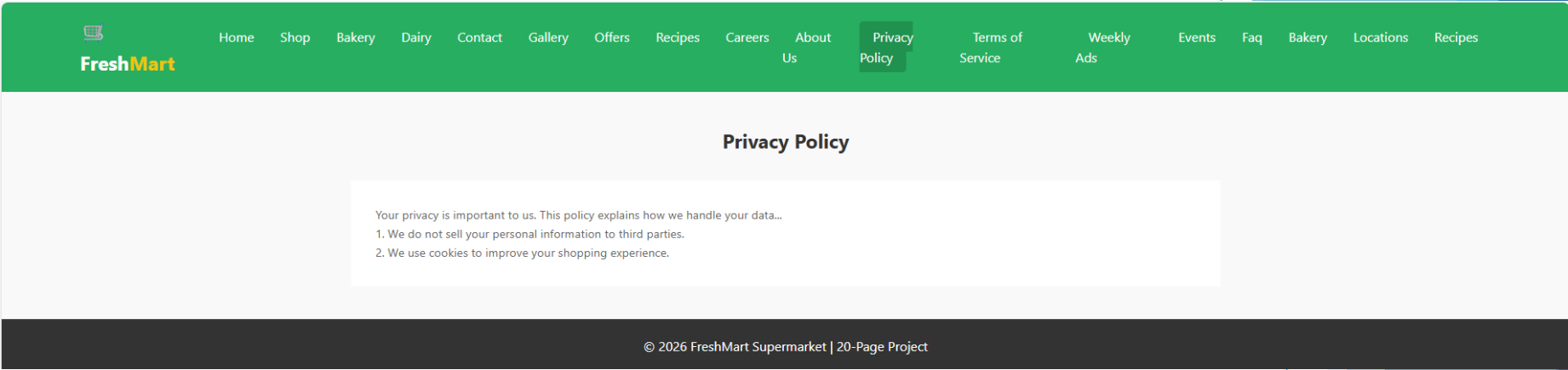
Our Story

Founded in 1995, FreshMart has been serving the community with the highest quality organic produce and local goods. Our mission is to provide affordable, healthy food for every family.
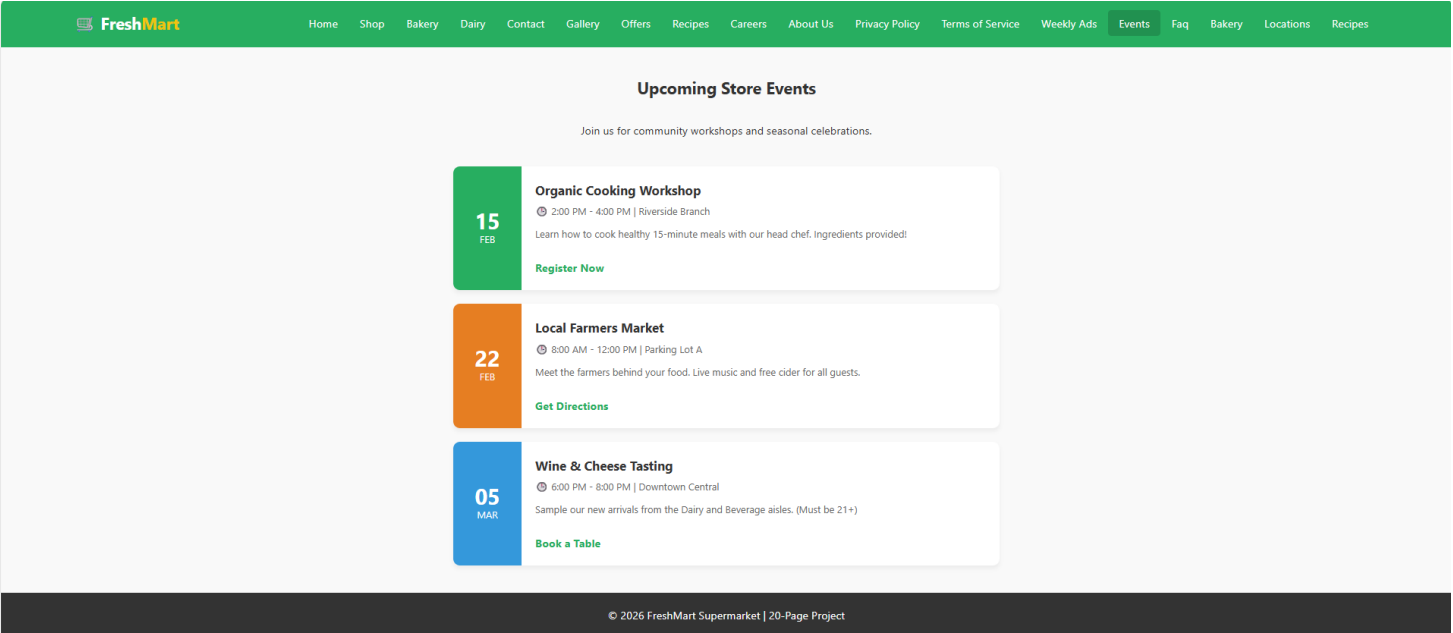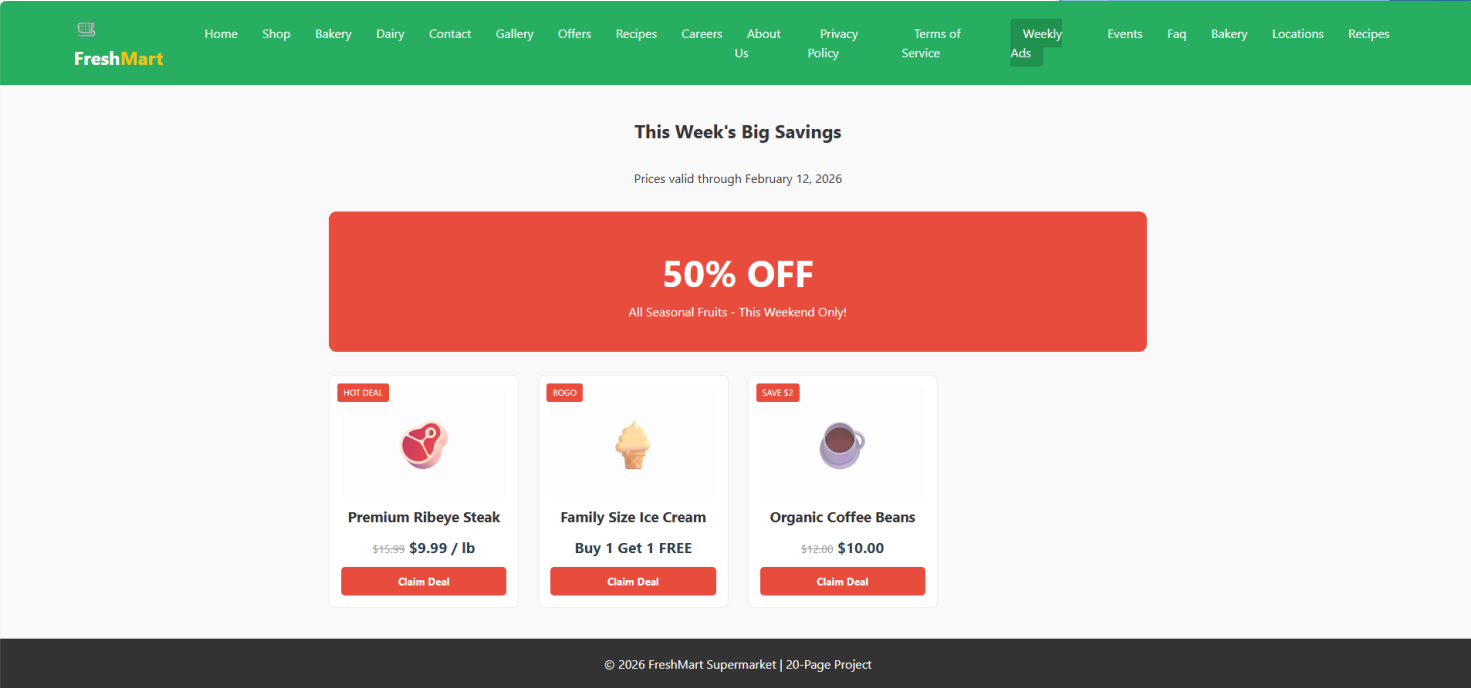
We work with over 50 local farmers to ensure that what ends up on your table is as fresh as it can be.

Lab Time

Privacy Policy

Your privacy is important to us. This policy explains how we handle your data...
1. We do not sell your personal information to third parties.
2. We use cookies to improve your shopping experience.

© 2026 FreshMart Supermarket | 20-Page Project

Terms & Conditions

Last Updated: February 2026

1. Introduction

Welcome to FreshMart. By accessing our website and using our services, you agree to comply with and be bound by the following terms and conditions.

2. Grocery Delivery

Delivery times are estimates and may vary due to traffic or weather conditions. We require a minimum order of $20.00 for all delivery services.

3. Pricing and Availability

While we strive for accuracy, errors in pricing or product descriptions may occur. FreshMart reserves the right to cancel orders arising from such errors.

4. Return Policy

Perishable items (like milk, meat, and vegetables) must be returned within 24 hours of purchase with a valid receipt for a full refund.

5. User Conduct

Users agree not to use this website for any unlawful purpose or any purpose prohibited under this clause.

© 2026 FreshMart Supermarket | 20-Page Project

Lab Time

This Week's Big Savings

Prices valid through February 12, 2026

50% OFF

All Seasonal Fruits - This Weekend Only!

HOT DEAL

Premium Ribeye Steak

$15.99 **$9.99 / lb**

Claim Deal

BOGO

Family Size Ice Cream

Buy 1 Get 1 FREE

Claim Deal

SAVE $2

Organic Coffee Beans

$12.00 **$10.00**

Claim Deal

Upcoming Store Events

Join us for community workshops and seasonal celebrations.

15 FEB

Organic Cooking Workshop

2:00 PM - 4:00 PM | Riverside Branch

Learn how to cook healthy 15-minute meals with our head chef. Ingredients provided!

Register Now

22 FEB

Local Farmers Market

8:00 AM - 12:00 PM | Parking Lot A

Meet the farmers behind your food. Live music and free cider for all guests.

Get Directions

05 MAR

Wine & Cheese Tasting

6:00 PM - 8:00 PM | Downtown Central

Sample our new arrivals from the Dairy and Beverage aisles. (Must be 21+)

Book a Table

Lab Time

Frequently Asked Questions

Do you offer home delivery?
Yes! We deliver within a 10-mile radius of any store.

What are your store hours?
We are open from 7:00 AM to 10:00 PM every day.

FreshMart

Home Shop Bakery Dairy Contact Gallery Offers Recipes Careers About Us Privacy Policy Terms of Service Weekly Ads Events Faq Bakery Locations Recipes

Our Store Locations

🏙️ Downtown Central
123 Main Street, Suite 100
New York, NY 10001
Hours: 7am - 11pm
📞 (555) 123-4567

View on Map

🌳 Riverside Branch
456 Park Avenue
Brooklyn, NY 11201
Hours: 8am - 10pm
📞 (555) 987-6543

View on Map

🛍️ Westside Mall
789 Sunset Blvd
Queens, NY 11101
Hours: 24 Hours
📞 (555) 000-1111

View on Map

Lab Time

✅

Message Sent Successfully!

Thank you for reaching out to FreshMart. Our team will get back to you within 24 hours.

Return to Home

404

🥛

Oops! Clean up on aisle four.

It looks like the page you are looking for has been moved or eaten.

Back to Home   Start Shopping