📘 E-Cell IIIT-Delhi | Developer Documentation

> A modern website built for E-Cell IIIT-Delhi using **Next.js 13 App Router**, **Tailwind CSS**, and custom components with modular structure.

E-Cell (Entrepreneurship Cell) of IIITD is a Official Student Body consist of remarkable group of students who are passionate about fostering an entrepreneurial spirit amongst their peers. In a world where startups are increasingly important for driving innovation and solving problems.

Table of Contents

🚧 Project Overview

This is a [Next.js](#) web application bootstrapped with `create-next-app`, built with a **component-based design**, **file-based routing**, and **modular APIs**. It uses **Yarn** as the package manager.

⚙ Tech Stack

Category	Technology
Framework	[Next.js 13+](#)
Styling	[Tailwind CSS](#)
Animations	`framer-motion`, `tailwindcss-animate`
UI Primitives	`@radix-ui/react-*`, `lucide-react`
Icons	`react-icons`, `lucide-react`
Mail	`nodemailer` for server-side email sending
Utils	`clsx`, `tailwind-merge`, `class-variance-authority`
Language	TypeScript
Deployment	Vercel (testing)

📌 Features of E-Cell IIIT-Delhi Website

The official E-Cell IIIT-Delhi website is built not only for performance but also for longevity, modularity, and ease of contribution. Here's a list of all key features integrated into the platform:

📟 UI/UX & Design

- ✅ **Modern UI** with clean design patterns
- 🎨 **Typography & Buttons** with Tailwind utility styles
- 🌈 **Uniform Theme, Padding, Margins** across pages
- 🧊 **Glassmorphism NavBar** with Submenu, Hamburger, and Mobile Navigation
- 🔄 **Responsive Design** for mobile, tablet, laptop, and desktop
- 🔍 **SEO Optimization**

- 🎬 **Video Background Hero Section**
- 📱 **Dynamic Loader Page**
- 🚫 **Custom 404 Not Found Page**
- 🧩 **Temp Page Templates** (Temp1 – Temp4) for future use

🧠 Structure & Modularity

- 🗂 **One Data Organisation Principle**
- 🔄 **OPPS-Optimised JSON Structure**
- 🧾 **Coding Documentation with Manual + Orientation Sessions**
- ⚙ **Secure File System** with static exports
- 🛡 **Safe Configuration aligned with IT-Team@IIITD**
- 📦 **Future-Proof Component Templates**

🧩 Page Features

- ⭐ **Intro, About Us, and Highlights Section**
- 🧑‍💼 **Team Page** with organized JSON, social links & roles
- 🖼 **Gallery Section** with collage layout, modals, and download facility
- 🚀 **Initiatives Page** with expandable modal layout
- 🧪 **Testimonials Page**
- 🎥 **Video Gallery with Custom Thumbnails**
- 🧾 **FAQs Page** + Linked FAQ Section
- 📣 **Contact Us Page** (mail service + map + socials)
- 🧭 **Resources Page + Structured Resources Section**
- 🔗 **Important External Links Section**

💡 E-Cell & IIIT-Delhi Specific Additions

- 🧠 **Ihub Anubhuti Section**
- 🧪 **IIC Achievements Section**
- 🧪 **Incubation & Innovation Section**
- 🏛 **About IIITD Section**
- 👥 **Invitees Section**
- 🤝 **Collaborations Section**

🛠 Interactive & Services

- 🧾 **Newsletter Service Integration**
- 📧 **Email API (via Nodemailer)**
- 📡 **Mail Service for Contact Inquiries**
- 🔀 **Interactive Team Archive Viewer**
- 🗃 **Clickable Social Grids in Footer & Pages**
- 🧠 **Interactive Gallery & Modal Handling**
- 🧠 **Interactive Initiative Detail Modals**

- 🧠 **Interactive Team Cards with Year-wise Sorting**

🌟 Exclusive Properties

- 🧑‍🎨 **Custom 3D Artpack** – Designed in-house, exclusive to E-Cell IIIT-Delhi

---

📂 Full Project Structure

```
ecell/
├── components.json
├── next-env.d.ts
├── next.config.js
├── package.json
├── postcss.config.js
├── README.md
├── tailwind.config.ts
├── tsconfig.json
├── yarn.lock
├── public/
│   ├── about/            # About images
│   ├── art/              # Art asset images (1.png – 12.png)
│   ├── Events/           # Event images
│   ├── gallery/          # Gallery section images
│   ├── img/              # Logos, images (e.g., 'E-Cell(1).png')
│   ├── initiatives/      # Visuals for initiatives
│   ├── team/             # Team asset images
│   ├── testimonials/     # Testimonials asset images
│   ├── Videos/           # Video Thumbnails (e.g., 1.avif)
│   ├── bg.mp4            # Background video
│   ├── logo.png
│   ├── web-app-manifest-192x192.png
│   └── web-app-manifest-512x512.png
├── src/
│   ├── app/              # Next.js App Router directory
│   │   ├── api/          # API routes
│   │   │   ├── sendEmail/route.js
│   │   │   └── subscribe/route.js
│   │   ├── about/page.tsx
│   │   ├── contact/page.tsx
│   │   ├── gallery/page.tsx
│   │   ├── initiatives/page.tsx
│   │   ├── resources/page.tsx
│   │   ├── team/page.tsx
│   │   ├── temp1–4/       # Temporary or demo pages
│   │   ├── favicon.ico, apple-icon.png
│   │   ├── globals.css    # Global Tailwind styles
│   │   ├── layout.tsx     # Root layout
│   │   ├── loading.tsx    # Loading state component
│   │   ├── not-found.tsx  # Custom 404
│   │   └── page.tsx       # Landing page
```

```
|       ├── components/        # All reusable components
|       |   ├── ui/            # Base UI (button, input, label, etc.)
|       |   ├── magicui/       # Effects like spinning text
|       |   ├── About.tsx, Hero.tsx, Events.tsx, TeamShowcaseCard.tsx, etc.
|       ├── data/              # Static data for components
|       |   ├── Faqs.json, TeamMembers.json, Gallery.json, About.json etc.
|       └── lib/
|           └── utils.ts       # Custom utility functions
```

📦 package.json Summary (Yarn)

```
yarn install   # Install all packages
yarn dev       # Run local dev server
yarn build     # Build for production
yarn start     # Start production server
yarn lint      # Run ESLint
```

🌐 Development Server

```
https://github.com/BhatiRishabh/ecell
cd ecell
yarn install
yarn dev
```

Visit http://localhost:3000

🌐 Production mode

```
yarn build
yarn start
```

Visit http://localhost:3000

📨 Email API

> A Google **App Password** is a 16-character one-time password you create in your Google Account to let less-secure apps (like nodemailer email clients) access your Gmail without sharing your main Google password. To generate one, sign into your Google Account, go to **Security** → enable **2-Step Verification** if it's not already on, then select **App passwords**. From there choose the app (or "Other (Custom name)") and device, click **Generate**, and Google will show a 16-character code — copy it (remove any spaces) and paste it into your app's credential store (for example `EMAIL_PASS` in your

> `.env.local`). Treat this code like a password: store it securely, and revoke it from the same **App passwords** page whenever you no longer need it. Note: App Passwords aren't available if your account uses Advanced Protection or if 2-Step Verification is disabled.

- API routes: `src/app/api/sendEmail/route.js src/app/api/subscribe/route.js`

- Powered by: **nodemailer**

- Configure credentials in `.env.local`

```
EMAIL_USER=<EMAIL>
EMAIL_PASS=<PASSWORD>
```

🖼 Image / Video Assets

Organized neatly in `/public/`:

- `/art` → Themed poster assets
- `/gallery` → Images for the gallery section
- `/Events`, `/initiatives`, `/Videos`, `/img`, `/team`, `/testimonials`, `/about` → Specific use-case images and videos
- `bg.mp4` → Background video

📥 Data Entry & Update Guide

All dynamic content for the E-Cell IIIT Delhi website is managed through structured JSON files in `src/data/`. Below is a guide to safely edit each section.

◆ 1. FAQs (`Faqs.json`)

File Path: `src/data/Faqs.json`

```
[
  {
    "question": "What is E-Cell IIIT Delhi?",
    "answer": "E-Cell is the Entrepreneurship Cell of IIIT Delhi..."
  }
]
```

➡ How to Add:

- Append a new object with `question` and `answer`.

◆ 2. Gallery (`Gallery.json`)

File Path: `src/data/Gallery.json`

```json
[
  { "id": 1, "title": "Bid Buzz" }
]
```

➡ **How to Add:**

- Use **unique** numeric `id`s.
- And put images in folder `public/gallery` mapped to id.
- eg: 1.png mapped to `{ "id": 1, "title": "Bid Buzz" }`
- Keep titles meaningful.
- Put Compressed Images Only to minimize website load.

◆ 3. Initiatives (`Initiatives.json`)

File Path: `src/data/Initiatives.json`

```json
{
  "id": "founders-talk",
  "name": "Founders Talk",
  "desc": "Real stories from startup founders...",
  "longDesc": "Detailed event overview...",
  "img": "/initiatives/founders-talk.png",
  "link": "/founders-talk"
}
```

➡ **How to Add:**

- Use a slug-style `id` (lowercase, hyphenated).
- Add clear `desc` and detailed `longDesc`.
- Ensure image paths exist in `/public/initiatives/`.

◆ 4. External Links (`Links.json`)

File Path: `src/data/Links.json`

```json
{
  "title": "Startup Registration",
  "url": "https://ecell-iiitd.in/startup-form",
  "desc": "Got a startup idea? Let`s help you scale it!"
}
```

➡ **How to Add:**

- Ensure URLs are correct and secure (`https`).
- Avoid duplicate `title` entries (currently repeated entries exist — clean up duplicates).

◆ 5. Team Members (`TeamMembers.json`)

File Path: `src/data/TeamMembers.json`

```json
{
  "2024": [
    {
      "name": "Aarav Mehta",
      "role": "President",
      "img": "/team/10.png",
      "linkedin": "https://linkedin.com/in/aarav",
      "instagram": "https://instagram.com/aarav",
      "email": "aarav@ecell.iiitd.ac.in"
    },
    {
      "name": "Ishita Kapoor",
      "role": "Vice President",
      "img": "/team/11.png",
      "linkedin": "https://linkedin.com/in/ishita",
      "instagram": "https://instagram.com/ishita",
      "email": "ishita@ecell.iiitd.ac.in"
    }
    // more members...
  ],
  "2023": [
    {
      "name": "Rohan Das",
      "role": "President",
      "img": "/team/2023-10.png",
      "linkedin": "https://linkedin.com/in/rohan",
      "instagram": "https://instagram.com/rohan",
      "email": "rohan@ecell.iiitd.ac.in"
    }
    // more from 2023...
  ]
}
```

➡ **How to Add:**

- Make Data Organise Yearwise as shown
- Use valid image paths under `/public/team/`.
- Include professional social handles.
- Put Compressed Images Only to minimize website load.

◆ 6. Testimonials (`Testimonials.json`)

File Path: `src/data/Testimonials.json`

```
{
  "name": "Ishita Sharma",
  "role": "Founder, StartupX",
  "image": "/testimonials/1.png",
  "text": "E-Cell helped me go from pitch to product..."
}
```

➡ How to Add:

- Keep `text` authentic and first-person.
- Store images in `/public/testimonials/`.

◆ 7. Video Gallery (`VideoGallery.json`)

File Path: `src/data/VideoGallery.json`

```
{
  "id": "iydMv_BAZJE",
  "title": "E-Summit 2024 Recap",
  "thumbnail": "/Videos/1.avif"
}
```

➡ How to Add:

- Use YouTube video `id` (not full URL).
- Add custom `thumbnail` path from `/public/Videos/`.
- **Avoid duplicating the same video object multiple times.** Each video should be listed only once unless showcasing variations.
- Put Compressed Images Only to minimize website load.

◆ 8. About (`About.json`)

File Path: `src/data/About.json`

```
{
    "name": "IIC Ranking ",
    "desc": "For the second year in a row, IIIT Delhi has secured the
prestigious 4-star rating in the Institutional Innovation Council (IIC)
rankings by AICTE, the highest distinction awarded this year! This
achievement cements IIITD`s position as a leading hub for innovation and
entrepreneurship. With a year of remarkable events, pioneering initiatives
by IIC, and the relentless efforts of E-Cell, the institute has
continuously fostered a thriving ecosystem for cutting-edge research,
startups, and industry collaborations. This recognition reflects the
```

/

```
    ingenuity, dedication, and forward-thinking mindset that drive IIIT Delhi`s
    commitment to excellence.",
        "img": "/about/6.png",
        "link": "https://iic.mic.gov.in/assets/html/index.html"
    },
```

➡ How to Add:

- Use a slug-style `id` (lowercase, hyphenated).
- Add clear `desc` and detailed `longDesc`.
- Ensure image paths exist in `/public/about/`.

✅ Best Practices

- ✅ Validate JSON with [JSONLint](#) before pushing changes.
- 🚫 Avoid duplicates (title, id, or URLs).
- 📂 Ensure referenced image/video paths exist in the `/public` folder.
- 🔑 Use HTTPS links and verified handles.
- 🧹 Keep formatting clean and consistent (2 spaces indent preferred).
- Put Compressed Images Only to Minimize website load.

🔧 Development Guidelines

Code Style

1. Use Arrow Functions in `.tsx` files
2. Use `<> </>` or `use client` declaration for fragmentation error
3. Follow UI Guidelines Seriously to maintain Consistency
4. Use `@/` alias for imports (maps to `app/` directory) Example: `import { InfiniteMovingCards } from '@/components/ui/infinite-moving-cards'`
5. Never delete code portions - comment them instead
6. Config files: Don't modify unless you understand their purpose

Best Practices

1. Always Refer to [Tailwind Docs](#) and [NextJS Docs](#)
2. Be aware of `.gitignore` contents
3. During git conflicts, always attempt to merge
4. Never copy-paste - append or modify instead
5. Map directories with Navbar links
6. Follow layout patterns to reduce redundancy
7. Debug Dark Mode when using libraries
8. Document complex logic
9. Never use CLI Install for UI/Libs/Etc. - use manual installation
10. Maintain accessibility standards
11. Implement responsive design

12. Involves a lot of icon and other libraries with the regular Node.js folder (contact developers for exact info about them)

UI/Layout Guidelines

1. Structure top and horizontal alignment using `bg-black text-white font-sans min-h-screen sm:px-24 px-12 pt-40`
2. Follow responsive design principles
3. Ensure dark mode compatibility
4. You can make copy of templates and make changes
5. Test across different screen sizes
6. General Header with Title of Any Page Template

```
  {/* Header */}
      <div className="flex flex-col-reverse lg:flex-row items-center
justify-between gap-12">
        <div className="w-full lg:w-1/2 flex flex-col gap-4">
          <h1 className="text-4xl md:text-6xl font-extrabold bg-gradient-
to-br from-purple-400 via-pink-500 to-red-400 bg-clip-text text-transparent
leading-tight drop-shadow-lg">
            Gallery of Impact
          </h1>
          <p className="text-lg md:text-xl text-gray-300">
            Dive into our vibrant visual universe, from events and posters
to raw creativity at E-Cell, IIIT Delhi.
          </p>
        </div>
        <div className="w-full lg:w-1/2 flex justify-center">
          <Image
            src="/art/5.png"
            alt="Gallery Artwork"
            width={400}
            height={400}
            className="w-[18rem] md:w-[22rem] lg:w-[26rem] h-auto object-
contain drop-shadow-2xl hover:scale-105 transition-transform duration-500"
          />
        </div>
      </div>
```

7. General Button Template

```
<Link href="/about">
<button className="p-[3px] relative">
  <div className="absolute inset-0 bg-gradient-to-r from-blue-500 to-
purple-500 rounded-full" />
  <div className="px-8 py-2  bg-black rounded-full font-bold relative group
transition duration-200 text-white hover:bg-transparent">
    Learn More
```

```
      </div>
    </button></Link>
```

⚠ Important Notes

IT-Team@IIITD Guidelines

1. No external input collection through website
2. All website export files must be static
3. Limited sudo access
4. Don't interfere with other users' files
5. Don't share sensitive data and access

Configuration Requirements

1. `next.config.mjs` settings:

```
  {
    output: 'export',     // Enable static export
    trailingSlash: true,  // Ensures trailing slash in URLs
    images: {
      unoptimized: true,  // Avoid IIITD server optimization errors
    }
  }
```

Development Notes

1. Be cautious with config files
2. Never delete code - comment it out instead
3. Review commented portions and Exp page for future use
4. Follow arrow function syntax in TSX
5. Use layout patterns to reduce redundancy
6. Implement proper fragmentation handling
7. Test dark mode compatibility
8. Use proper directory mapping
9. Follow import alias conventions
10. Always Do Deployment test at Localhost before Deployment and confirm safe run before pushing into the git

⚒ Troubleshooting

Common Issues

1. Fragmentation errors: Use `<> </>` or `use client` declaration
2. Dark mode issues: Test thoroughly when using new libraries
3. Directory mapping: Ensure proper alignment with Navbar
4. UI Library conflicts: Use manual installation instead of CLI

5. Use `'` for `'` & `"` for `''`

Server Access

1. VPN connection issues: Verify credentials and network
2. SSH access problems: Check username and IP
3. File permission issues: Work within user permissions

👥 Team & Credits

Built by: KintsugiDevStudio -- Siddhant Bali & Rishabh Bhati

📞 Contact & Support

For technical issues or access requests:

1. Lab Administration
2. IT Support: IT-Team@IIITD
3. Web Administration

🔁 Version Control

- Main repository: ECELL
- Archive: Contains previous versions for reference

📚 Learning Resources

- Next.js Docs
- Tailwind Docs
- TypeScript Handbook

Last Updated: June 2025 Maintained by E-Cell IIIT-Delhi