

Aniket Vilas Panchal

Full Stack Developer | .NET | Node.js | AWS | Next.js | Django

Hi! I'm **Aniket Panchal**, a Passionate **Full-Stack Developer** and **Cybersecurity Enthusiast** who thrives on turning innovative ideas into Scalable, Intuitive, and High-Impact Digital Products.

With a strong foundation in both **Frontend** and **Backend Development**, I specialize in building Modern, Scalable Web Applications using stacks like **MERN**, **MEAN**, **.NET**, and **Django**. My project experience spans **Social Media Platforms**, **Real-Time Multiplayer Games**, **Enterprise Dashboards**, and **Personal Portfolio Websites**.

I combine Creative Design with Technical Precision—whether crafting Sleek UIs with **React**, **Next.js**, or **Angular**, developing Robust APIs using **Node.js**, **.NET Web API**, or **Django**, or managing Databases like **MongoDB**, **PostgreSQL**, and **SQL Server (SSMS)**.

I'm also proficient in **DevOps** and **Cloud Technologies**, including **Docker**, **Git**, **Firebase**, and **AWS EC2**, while continuously deepening my expertise in **Linux Systems**, **DevOps Pipelines**, and **Cybersecurity** Practices.

Outside of tech, I enjoy creating **Digital Art**, exploring **Game Development in Unity**, and contributing to **Open-Source Projects**. I'm always excited to collaborate on Innovative Projects or join Teams where I can **Learn, Build, and Make a Real Impact**.



Projects Section:



FinShark - Financial Data Dashboard

React + .NET Web API



GitHub: <https://github.com/babanigit/Finshark>



Live Demo: <https://finshark-production.up.railway.app/>



Project Description:

- **FinShark** is a Modern **Financial Dashboard** that lets users **Search**, **Track**, and **Manage** Company Financials with ease.
- It integrates the **FMP (Financial Modeling Prep) API** to fetch **Live Stock Data**, including **Company Profiles**, **Income Statements**, **Balance Sheets**, and **Historical Dividends**.
- The app ensures **Secure Token-Based Authentication**, allowing only **Authorized Users** to manage **Personalized Portfolios** through protected **CRUD Operations**. Designed with **React (TypeScript)** for the **Frontend** and a **.NET Core Web API** for the **Backend**, FinShark is optimized for **Performance**, **Security**, and **User Experience**.








Tech Stack:

- **Frontend:** React.js (TypeScript)
- **Backend:** ASP.NET Core Web API
- **Database:** PostgreSQL
- **Authentication:** Token-Based Auth (JWT)
- **External API:** FMP Stock Market API
- **Deployment:** Dockerized App deployed on AWS EC2 and Railway



Key Features:

-  **Real-Time Company Search and Financial Data Tracking**
-  **View Income Statements, Balance Sheets, and Dividends**
-  **Secure Login System and User Authentication**
-  **Manage Personalized Portfolios with CRUD Operations**
-  **Dockerized Deployment for Scalability and Portability**



RunGroop RunnersApp - Social Platform for Runners

.NET MVC Web App



GitHub: github.com/babanigit/RunGroop_RunnersApp



Live Demo: <https://runggrooprunnersapp-production.up.railway.app/>



Project Description:

- **RunGroop** is a **Social Web Platform** built for **Runners** to **Discover Clubs**, **Schedule Events**, and **Connect with Fellow Athletes** in their area.
- It is developed using **ASP.NET Core MVC (v9)** with a **PostgreSQL Backend**, and features a **Clean, Minimal UI** designed with **Bootstrap CSS** for a seamless **User Experience**.
- The app includes **Secure JWT-Based Authentication**, **Cloudinary** for **Media Uploads**, and uses **Entity Framework Core** for **ORM** and **Database Management**. It also features an **Admin Panel** for **Content Management**.









Tech Stack:

- **Frontend:** Razor Pages (ASP.NET MVC Views)
- **Backend:** ASP.NET Core MVC 9
- **Database:** PostgreSQL (via EF Core + Npgsql)
- **Authentication:** ASP.NET Identity
- **Media:** Cloudinary (Image Uploads)
- **Deployment:** AWS EC2 or Railway with Docker
- **Others:** JWT, Dependency Injection, MVC Pattern



Key Features:

-  **Discover & Join Running Clubs** by Location
-  **View & Connect** with Other Athletes
-  **Create and Manage Club Events**
-  **Upload Profile and Club Images**
-  **Secure User Authentication & Role-Based Access**

-  **Admin Dashboard** for Managing **Users, Clubs, and Events**

Tic-Tac-Toe Game - Real-Time Web Game with Chat Integration

MERN Stack + Steam Chat

 **GitHub:** <https://github.com/babanigit/Tic-Tac-Toe-Game>

 **Live Demo:** <https://tic-tac-toe-game-production.up.railway.app/>








Project Description:

- A **Full-Stack Real-Time Multiplayer Tic-Tac-Toe Game** built using the **MERN Stack**, designed for **Smooth and Interactive Gameplay** between users.
- The application integrates the **Stream Chat API** for **Real-Time Messaging** and **WebSocket-Based Communication**, significantly enhancing the **Multiplayer Experience**.
- The platform features **JWT-Based Authentication, Session Handling, Game Room Creation** via Unique Codes, and Automated **Win/Draw Detection Logic**. Deployed using **Docker** on **AWS EC2**, it offers a **Scalable and Reliable Gaming Environment**.

Tech Stack:

- **Frontend:** React.js, Tailwind CSS
- **Backend:** Node.js, Express.js
- **Database:** MongoDB (via Mongoose)
- **Authentication:** JWT & Express Session
- **Real-Time & Chat:** Stream Chat API, WebSockets
- **Deployment:** Docker + AWS EC2

Key Features:

-  **User Authentication** with **Encrypted Password Storage**
-  **Create/Join Private or Public Games** via Unique Room Codes
-  **Real-Time Multiplayer Gameplay** using **WebSockets**
-  **Integrated Chat System** powered by **Stream Chat API**
-  **Game Logic** with **Win-Checking, Draw Detection, and Scoring**
-  **Responsive Design** for **Desktop & Mobile**
-  **Dockerized Deployment** for **Production Readiness**

To-Do App - Full-Stack Task Manager

MERN Stack + Redux

 **GitHub:** <https://github.com/babanigit/To-Do-App>

 **Live Demo:** in process









Project Description:

- A **Robust and Efficient Task Management Application** built using the **MERN Stack** (MongoDB, Express.js, React.js, Node.js), enhanced with **Redux** for **Global State Management** and **TypeScript** for **Type Safety**.
- The app allows users to **Create, Update, Delete, and Manage Tasks** through an **Intuitive UI** with **Real-Time Updates**. It supports **Dark/Light Theme Toggle**, **Priority-Based Task Sorting**, and **Cookie-Based JWT Authentication** for **Secure User Sessions**.
- The project is also equipped with **Docker Support** for seamless **Containerization** and **Git Pre-Commit Hooks** to enforce **Code Quality Standards**.

Tech Stack:

- **Frontend:** React.js (**TypeScript**), **Redux**, **Tailwind CSS**
- **Backend:** Node.js, **Express.js**
- **Database:** **MongoDB**
- **Authentication:** **JWT Cookie-Based Authentication**
- **Deployment:** **Dockerized App** hosted on **AWS EC2**

Key Features:

-  **Dark & Light Theme Toggle** for UI Customization
-  **Full CRUD Task Management**
-  **Priority-Based Task Sorting** (by Due Date / Status)
-  **Secure Authentication** using **Cookie-Based JWT**
-  **Robust Error Handling** on the Backend
-  **Docker Containerization** for Portability
-  **TypeScript** for Scalability & Maintainability
-  **Git Pre-Commit Hooks** for Code Quality Control

Music Classes Portfolio - Final Year Project for Music Class Enrollment System


MERN Stack Web App










 **GitHub:** <https://github.com/babanigit/Third-Year-MERN-Project>

 **Live Demo:** <https://third-year-mern-project.onrender.com/>

Project Description:

- **Music Classes Portfolio** is a **Full-Stack Web Application** developed as a **Final Year College Project**, designed to help users **Explore and Apply for Music Classes** through a **Modern, Responsive Interface**. Once a student applies, their details are submitted, and teachers can **Directly Connect** for further communication.
- The app is built using the **MERN Stack**, enhanced with **Redux** for **State Management**, and utilizes **Firebase** for **Google Authentication** and **Real-Time Capabilities**. Styled with **Tailwind CSS**, the interface is **Mobile-First** and fully **Responsive**. It also incorporates **JWT Cookie-Based Authentication** for **Secure Sessions**, and features an **Admin Dashboard** for managing class applications.

-  **Tech Stack:**
- **Frontend:** React.js, **Redux**, **Tailwind CSS**
 - **Backend:** Node.js, **Express.js**
 - **Database:** **MongoDB**
 - **Authentication:** **JWT Cookie-Based Auth**, **Google Sign-In via Firebase**
 - **Extras:** **Firebase Real-Time DB**, **Responsive Design**
 - **Deployment:** **Railway**

-  **Key Features:**
-  **Apply for Music Classes Online**
 -  **Teachers Receive Applicant Info Instantly**
 -  **Secure Login with Google and JWT Cookies**
 -  **Admin Dashboard** for Managing Class Entries
 -  **Fully Responsive, Mobile-First Design**
 -  **Real-Time Database** Support via **Firebase**
 -  **Sleek UI with Tailwind CSS** and Modern UX Patterns
 -  **Docker Containerization** for Portability


Fighting Dots - Multiplayer Game - Real-Time Online Game






Built with Node.js & Socket.IO



-  **GitHub:** <https://github.com/babanigit/Node.js-OnlineGame>
-  **Live Demo:** <https://node-js-onlinegame.onrender.com/>

Project Description

- A **Real-Time Multiplayer Fighting Game** developed using **Node.js** and **Socket.IO**, designed for **Interactive Online Battles** in a **Responsive, Real-Time Environment** powered by **WebSockets**. The game emphasizes **Smooth Performance**, **Low Latency**, and **Accurate Multiplayer Synchronization**.
- Built with **TypeScript** for **Scalability** and **Maintainability**, the game uses **Node.js** to handle **Real-Time Game Logic** and **Socket.IO** for **Seamless Bi-Directional Event Communication** between players.

-  **Tech Stack**
- **Runtime:** Node.js
 - **Communication:** Socket.IO (**WebSockets**)
 - **Language:** TypeScript
 - **Architecture:** **Event-Driven**, Multiplayer Game Logic

-  **Key Features**
-  **Real-Time Multiplayer Gameplay** using WebSockets
 -  **Battle Other Players** in an Interactive Fighting Arena
 -  **Fast & Responsive Player Movements and Attacks**
 -  **Bi-Directional Data Flow** with Low Latency via Socket.IO

-  **TypeScript-Based Server Logic** for Scalability & Clarity
 -  **Future-Ready Architecture** for Browser-Based Game Deployment
-

Internship Experience:

Angular Developer Internship





 **Markets Mojo – Stock Research Organization, Mumbai**

 **July 2024 – September 2024**

 https://github.com/babanigit/markets_mojo_v16 |

 <https://markets-mojo-v16.vercel.app/portfolio-plus/stock-investments>

Key Contributions:

-  **Migrated Legacy Angular Pages to Angular 16 with TypeScript**, significantly improving **Performance** and **Code Maintainability**
-  **Integrated Highcharts to Visualize Stock Data**, enhancing **Data Insight Delivery** by **30%**
-  Utilized **Chrome DevTools** for **Extensive Debugging and Profiling**, increasing **Development Efficiency**
-  Achieved **Pixel-Perfect UI Replication** using **Bootstrap**, and **Containerized** the entire project with **Docker** for seamless **Deployment**

Topics & Tech: Angular 16 · TypeScript · Bootstrap · Highcharts · Chrome DevTools · Docker

Closing & Contact

Thank you for taking the time to explore my portfolio.

I'm deeply passionate about building Scalable, Secure, and User-Centric Applications that solve real-world problems.

If my work aligns with your Vision, Team, or Project Goals, I'd be excited to connect and collaborate!

Feel free to reach out through any of the platforms below:

-  **Email:** aniket.vilas.panchal@gmail.com
-  **LinkedIn:** linkedin.com/in/aniket-vilas-panchal
-  **GitHub:** github.com/babanigit