<https://divasgupta.com/masterclass/>

WEBSITE Project

**I had the opportunity to work on a project during my web development internship at MMM Info Tech in Tirunelveli, where I built a website for St. Michael Church Vazhaithottam. The church wanted a platform where they could showcase information, events, and announcements for their community. My role was to handle the website's development from scratch.**

**For the frontend, I used HTML, CSS, and JavaScript to design a user-friendly and responsive website. I focused on creating a simple yet elegant layout, ensuring that the site would work well on both desktops and mobile devices. One of the key challenges was making sure the website was easy to navigate while still providing a visually appealing design that fit the church's brand.**

**After developing the site, I hosted it on GitHub Pages, which allowed the church to manage updates easily. In addition, I configured a custom domain for the site using GoDaddy, handling both DNS settings and domain registration. This step involved integrating GitHub's hosting platform with GoDaddy’s domain management to ensure that the site was accessible via the custom domain."**

**Overall, the project allowed me to apply my web development skills in a real-world setting, from frontend design to hosting and domain configuration. It also taught me how to manage client expectations and deliver a product that met both functional and aesthetic requirements."**

**Can you explain the technical challenges you faced while working on this project?**

* **Answer:** One challenge I encountered was integrating the HTTPS SSL certificate, which was crucial for the church’s online security. As I worked alone on the project, I took full responsibility for troubleshooting and testing the certificate integration to ensure secure communication. Another technical challenge was designing a responsive layout using Bootstrap that would look good on all devices, from desktops to mobile phones. I made sure the website delivered with pride by rigorously testing it across different platforms and browsers to guarantee a seamless user experience.

**Introduction**:

"My project is called **QuickCart**, and it’s an eCommerce website built using the **MERN stack**. MERN stands for **MongoDB, Express.js, React, and Node.js**. QuickCart lets users browse products, add items to a shopping cart, and place orders, just like any online shopping platform."

**Frontend**:=

"On the frontend, I used **React** to build the interface, so it’s a Single Page Application, which makes it fast and interactive. For navigation between pages, I used **React Router**, and to handle the shopping cart and other states, I used **React hooks** like useState and useEffect. The cart updates instantly when users add, remove, or change quantities, and I used **React Toastify** to give notifications like 'Item added to cart' so users get instant feedback."

**Backend**:

"For the backend, I used **Node.js** and **Express.js** to create a server that handles things like product data and orders. All the product and order information is stored in **MongoDB**. I used **Mongoose** to create models for both products and orders, so when users add items to their cart or place an order, that data is saved and managed efficiently in the database."

**Main Features**:

* **Browsing Products**:
  + "Users can search for products and see all the available items with details like price and ratings."
* **Shopping Cart**:
  + "They can add items to the cart, increase or decrease quantities, and remove items. The cart updates in real-time."
* **Placing Orders**:
  + "Once they finalize the cart, they can place an order, and the backend saves the order details to the database."

**Challenges & Future Plans**:

"I had to make sure the frontend and backend worked smoothly together, which was challenging at first, but I managed it with React hooks and proper API handling. In the future, I plan to add features like **user login, payment integration, and an admin panel** to manage products and orders."

**Conclusion**:

"QuickCart shows my ability to build a full-stack application from scratch using the **MERN stack**.

**"Will AI take over jobs?"** in an interview:

**Yes, AI will change jobs, but it’s not as simple as AI taking over all jobs.**

AI will automate repetitive and simple tasks, like data entry or basic customer service. This could reduce the need for some types of jobs, especially ones that involve doing the same thing over and over. But AI will also create new opportunities in areas like AI development, maintenance, and roles that require human creativity, decision-making, and emotional intelligence.

So, while some jobs might disappear, many new ones will be created, and many existing jobs will just change. People will need to learn new skills to stay relevant in this changing world.

**Project Overview:**

The project I worked on is a **Medicinal Plant Identification System** with an added **chatbot** feature. The system helps users do two things:

**Step-by-Step Explanation:**

1. **Plant Identification**:
   * When a user uploads an image of a plant, our system processes that image to **identify what plant it is**.
   * We built a **machine learning model** that can recognize different medicinal plants by analyzing the image.
   * The model was trained using a technology called **Transfer Learning** with a popular model called **Inception v3**, which is good at identifying images.
   * After the user uploads the image, the model predicts the plant name, and we display it to the user.
2. **Chatbot Feature**:
   * After identifying the plant, the user can ask questions about it, such as, "What are the health benefits of this plant?" or "How can this plant be used as medicine?"
   * We integrated **OpenAI's GPT-3** to create a chatbot that responds to these types of questions.
3. **Frontend (User Interface)**:
   * We designed a simple web interface where users can **upload an image** of a plant, see the **prediction**, and chat with the **chatbot** for further information.
   * The front end uses basic **HTML, CSS, and JavaScript** to make the website user-friendly.
4. **Backend (Processing the Image)**:
   * The backend, where all the processing happens, is built using **Flask**, a framework for building web applications in **Python**.
   * It handles the image processing and runs the machine learning model on the server, and also processes the questions asked to the chatbot.

Any Questions?

"Could you share some recent projects completed by previous interns that made a significant impact within the PBWMT division?

so that I too set similar goals during my internship to achieve a comparable level of impact."

To talk about my project "Student Guardian App." This app helps students manage their school profiles and stay updated with important announcements from teachers.

**Purpose:** The main goal of the app is to enhance communication between teachers and students. Traditional methods like emails can be less engaging, and using WhatsApp feels too informal for school-related communication. The Student Guardian App provides a dedicated platform that makes interactions more professional and effective.

**Key Features:**

1. **Profile Management:**
   * Students can create and edit their profiles, including their name, student ID, and contact details. This feature ensures that their information is always up to date.
2. **Announcements:**
   * Teachers can send messages and updates directly to students through the app. This ensures that students receive important news about classes or events promptly, reducing the chances of missing vital information.
3. **User-Friendly Design:**
   * The app is designed to be intuitive and easy to use. Even those who aren't tech-savvy can navigate it without difficulty, making it accessible for all students.

**Technical Details:**

* The Student Guardian App is built using **Flutter**, a framework that allows for creating apps for both Android and iOS from a single codebase.
* For data storage and retrieval, we used **Firebase Realtime Database**. This technology keeps data synced in real-time, so when a student updates their profile or a teacher sends an announcement, it instantly reflects in the app for everyone.

I think dell technologies is the big platform for me to start my career and the job description says that the company is in need of a employee with knowledge in aiml and development , and this is where iam a expert in which can be signified from my previous internship in mmm info tech wherein I made a fully functional responsive website from scratch and made both the client as well as the company happy

part from these iam a person who is determined and passionate to work in any environment

and also, iam a mix of hardworking as well as smart working person which is evident from my academic background, finally I also have a great desire to work in the dell technologies culture

so considering all those things you should hire me

Strengths

I always belive in the magic word “I can” which made me to have knowledge strength in various fields like aiml, and development , also iam a adaptable person who works well in teams which is evident from my participation in hackathons and collaborative projects

Also I have a passion for learning which makes me to expand my knowledge base in evolving tech landscape

Can You Concisely Differentiate AI ML:

 **AI (Artificial Intelligence)**:

* Broad field focused on creating systems that can perform tasks requiring human-like intelligence (e.g., reasoning, decision-making, language understanding).
* Includes a wide range of techniques like rule-based systems, search algorithms, and robotics, in addition to machine learning.

 **ML (Machine Learning)**:

* A subset of AI that focuses on building algorithms that allow systems to **learn from data** and improve performance over time without explicit programming

**Technical Difference between AI, ML, DL & Related Fields**

**1️⃣ Artificial Intelligence (AI)**

* **AI** is the **broad science** of creating systems that can perform tasks which typically require **human intelligence**.
* It encompasses **reasoning, learning, problem solving, perception, and even creativity**.

**➡️ Technically:**

* AI involves rule-based systems (expert systems), search algorithms, logic, and planning.
* Examples include **rule-based medical diagnosis systems**, chess-playing programs using **minimax search**, and knowledge graphs.

AI can exist **even without data-driven learning**. For example, an expert system with hard-coded rules for credit approval is AI but **not ML**.

**2️⃣ Machine Learning (ML)**

* **ML** is a **subset of AI**, where we build algorithms that allow machines to **learn patterns from data** and make decisions or predictions.

**➡️ Technically:**

* Uses **statistical techniques** to learn from historical data.
* Examples include:
  + **Supervised Learning:** Regression, Decision Trees, SVM, Neural Networks
  + **Unsupervised Learning:** Clustering (K-Means, GMM), Dimensionality Reduction (PCA, LDA)
  + **Reinforcement Learning:** Learning by rewards & penalties

**Mathematically**, ML minimizes a **loss function** over data samples to generalize to unseen data. E.g.,

min⁡θ1N∑i=1NL(fθ(xi),yi)\min\_\theta \frac{1}{N} \sum\_{i=1}^N L(f\_\theta(x\_i), y\_i)θmin​N1​i=1∑N​L(fθ​(xi​),yi​)

where θ\thetaθ are model parameters.

**3️⃣ Deep Learning (DL)**

* **DL** is a **subset of ML**, using multi-layered **artificial neural networks** to model complex non-linear relationships.

**➡️ Technically:**

* Uses architectures like:
  + **CNNs (Convolutional Neural Networks)** for images
  + **RNNs, LSTMs, Transformers** for sequences
* Learns hierarchical features. For instance, a CNN learns edges in early layers, shapes in deeper layers.

**Optimization is via backpropagation**, adjusting weights to minimize loss.

Example from your projects:

* In **BG-Vanish**, you used **U²-Net**, a deep CNN architecture, to generate saliency masks for background removal.

**🧭 Related Subfields (Technically Linked to ML/DL)**

**📸 Computer Vision (CV)**

* Focused on enabling machines to **interpret visual data**.
* Uses ML/DL models (like CNNs, Mask R-CNN, U²-Net) to do:
  + **Image classification** (Is this a cat or dog?)
  + **Object detection** (Where is the dog in this image?)
  + **Segmentation** (Which pixels belong to the dog?)
* Example: In **BG-Vanish**, U²-Net segments foreground from background.

**🗣 Natural Language Processing (NLP)**

* Enables machines to **understand & generate human language**.
* Uses embeddings (Word2Vec, BERT), RNNs, Transformers.
* Example: In **Mithran App**, you used **Dialogflow**, which internally builds NLP models to understand farmer queries.

**🎮 Reinforcement Learning (RL)**

* Unlike supervised ML, RL learns by **interacting with an environment** and maximizing rewards.
* Formally uses **Markov Decision Processes (MDPs)**:

π∗=arg⁡max⁡πE[∑t=0∞γtR(st,at)]\pi^\* = \arg\max\_\pi \mathbb{E} \left[ \sum\_{t=0}^\infty \gamma^t R(s\_t, a\_t) \right]π∗=argπmax​E[t=0∑∞​γtR(st​,at​)]

* Used in robotics, games (AlphaGo).

**⚙️ Summarized Technical Hierarchy**

pgsql

CopyEdit

AI

│

├── ML (learns from data)

│ ├── Supervised (SVM, RF, NN)

│ ├── Unsupervised (K-Means, PCA)

│ └── RL (learns via trial-reward)

│

└── DL (ML using deep neural nets)

├── CNNs for images

├── RNNs/LSTMs for sequences

└── Transformers for language

**🔬 Examples from Your Projects**

| **Project** | **Field** | **Details** |
| --- | --- | --- |
| **BG-Vanish** | CV + DL | Used U²-Net (deep CNN) for saliency maps and background removal |
| **Mithran** | NLP + DL | Used Dialogflow (sequence models + intent classification) for farmer chatbot |
| **GenOrNot** | ML | Used SVM + PCA + handcrafted features (LBP, FFT) for classifying AI-generated vs real images |

✅ **In short:**

* **AI** = broad goal of smart systems
* **ML** = learning patterns from data
* **DL** = neural networks learning complex features from large datasets
* With specialized areas like **CV, NLP, RL** applying these principles to specific data types.

Most asked questions in Interview.

Tell me about yourself.

Walk me through your resume.

How did you hear about this position?

Why do you want to work at this company?

Why do you want this job?

Why should we hire you?

What can you bring to the company?

What are your greatest strengths?

What do you consider to be your weaknesses?

What is your greatest professional achievement?

Tell me about a challenge or conflict you’ve faced at work, and how you dealt with it.

Tell me about a time you demonstrated leadership skills.

What’s a time you disagreed with a decision that was made at work?

Tell me about a time you made a mistake.

Tell me about a time you failed.

Why are you leaving your current job?

Why were you fired?

Why was there a gap in your employment?

Can you explain why you changed career paths?

What’s your current salary?

What do you like least about your job?

What are you looking for in a new position?

What type of work environment do you prefer?

What’s your work style?

What’s your management style?

How would your boss and coworkers describe you?

How do you deal with pressure or stressful situations?

What do you like to do outside of work?

Are you planning on having children?

How do you stay organized?

How do you prioritize your work?

What are you passionate about?

What motivates you?

What are your pet peeves?

How do you like to be managed?

Do you consider yourself successful?

Where do you see yourself in five years?

How do you plan to achieve your career goals?

What are your career aspirations?

What’s your dream job?

What other companies are you interviewing with?

What makes you unique?

What should I know that’s not on your resume?

What would your first few months look like in this role?

What are your salary expectations?

What do you think we could do better or differently?

When can you start?

Are you willing to relocate?

How many tennis balls can you fit into a limousine?

If you were an animal, which one would you want to be?

Sell me this pen.

Is there anything else you’d like us to know?

Do you have any questions for us?