



SMART INDIA
HACKATHON
2024

SIH'24

Problem Statement :

**Develop a Cloud-Integrated IoT
Alarm Clock with Dashboard
Integration**

*PS Category- **Hardware***

Team Name :

BYTE MECHANICS

TEAM MEMBERS

Ravi Raj

Aditya Garg

Ashu Anand

Aditya Mohan

Hrydya

Riddhi



Introducing the Cloud-Integrated Alarm Clock

Visualize the Flow:

STEP:1

AWS connection  via MQTT client ensures secure and reliable communication.

STEP:2

Raspberry Pi uses Python to manage alarm settings, user inputs, and display controls, with robust data storage and retrieval via AWS DynamoDB..

STEP-3 :WEB CONTROL PANEL

Users can manage alarms and settings through a user-friendly web interface, all powered by a Node.js backend


STEP-4 :FEATURE RICH CONTROLS-

Easily customize your wake-up experience with snooze, light, and buzzer settings on the web platform.

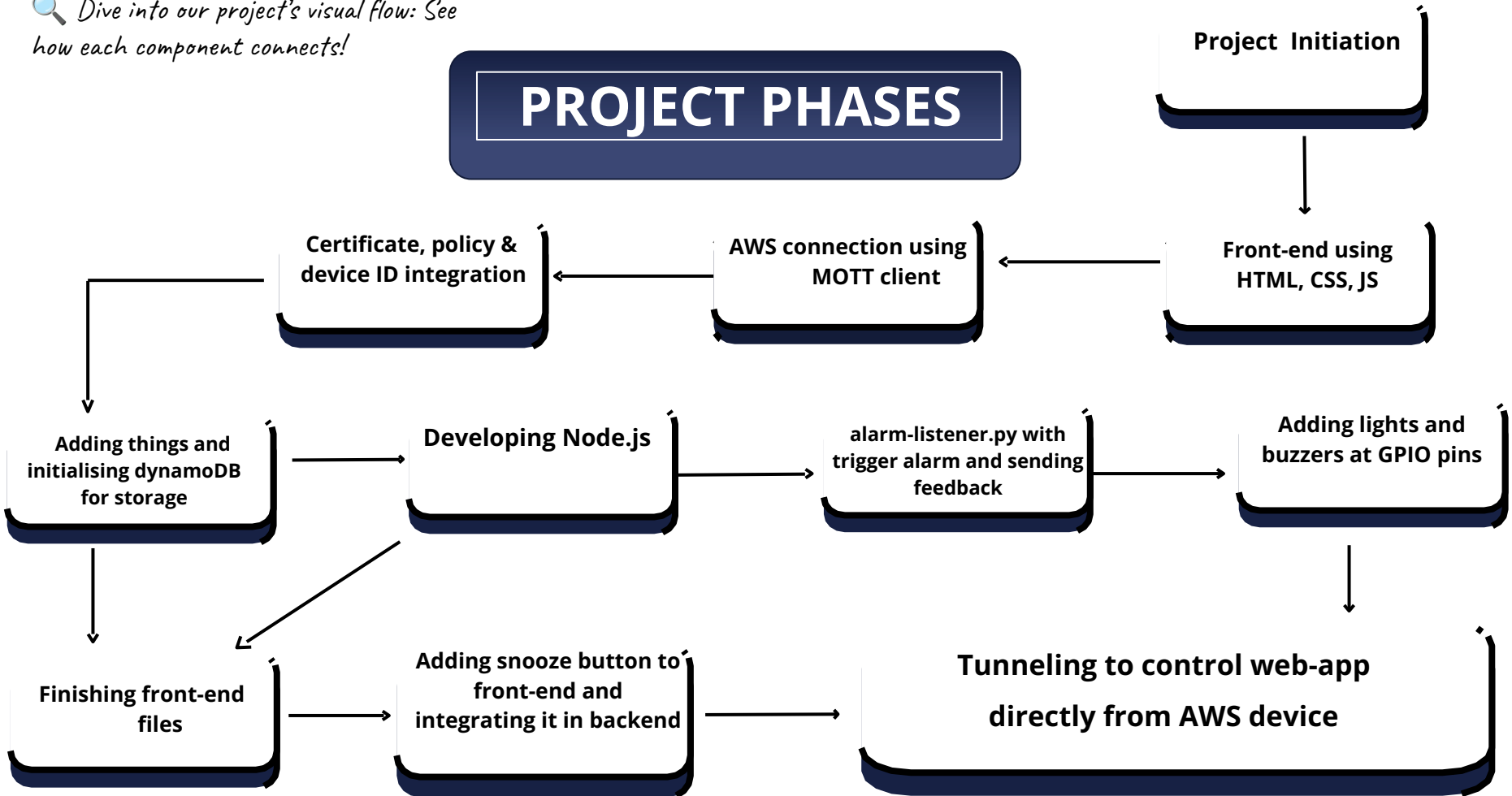


What's the Big Idea?

- **Smart & Simple:** Our solution reimagines the traditional alarm clock, integrating it with the cloud to bring convenience and control to your fingertips. With AWS IoT Core as the backbone, the clock does more than just wake you up—it connects you to a seamless, interactive experience.

 Dive into our project's visual flow: See how each component connects!

PROJECT PHASES



What makes us stand out?



01

U

User-Centric Interface:

The intuitive web platform and streamlined controls ensure that every interaction with your alarm clock is simple, efficient, and tailored to enhance your daily routine.

02

C

Cloud-Centric Design:

Our solution integrates deeply with the cloud, providing flexibility and control beyond what traditional alarm clocks can offer.

03

S

Secure and Reliable:

Leveraging AWS IoT Core, your data is securely stored and seamlessly synchronized, providing peace of mind that your settings are always protected and up to date.

04

R

Remote Control:

Set and manage your alarms from anywhere, ensuring your clock stays synchronized with your schedule, whether you're at home or on the go.

FEASIBILITY & VIABILITY

Tech Integration:

We're using HTML, CSS, Javascript for the sleek web interface, with Node.js and Python powering the backend. Our Raspberry Pi manages the LED and buzzer. It communicates with AWS DynamoDB via MATT

Why it works?

This combination of web, IoT, and cloud technologies is proven and practical, ensuring smooth integration and functionality.

Current Use:

Ideal for personal use or educational demos, demonstrating practical integration of web technologies, IoT, and cloud services.

Future Potential:

To stand out commercially, we'll add features like customizable alarm tones, smart home integration, and advanced user settings.

Customization and Expandability

We'll optimize AWS resources (such as DynamoDB and AWS Lambda) for cost-efficiency, plan for increased user load, and ensure robust data security and compliance with industry standards.



Curious about how each technology contributes to our solution? Let's break it down!

Learning and Skill Development: This project provides hands-on experience with a wide range of technologies, including web development (HTML, CSS, JavaScript), IoT hardware integration (Raspberry Pi, LED, buzzer), backend programming (Node.js, Python), cloud services (AWS DynamoDB), and messaging protocols (MQTT).

Practical Application of IoT: The project demonstrates a practical application of IoT principles, integrating hardware with software to create an interactive and automated solution. This can serve as a foundation for more advanced IoT projects or smart home systems.

Real-Time Data Management: Using AWS DynamoDB allows for efficient real-time data storage and management, enabling you to track alarms, user interactions, which can be useful for analytics or improving user experience.



Impacts & Benefits

Scalability and Flexibility: Leveraging cloud services and MQTT allows the system to be scalable and flexible. The backend can handle multiple devices and users if needed, and the MQTT protocol provides efficient, lightweight messaging for IoT environments.

Customization and Expandability: The project is highly customizable, allowing for further development, such as adding more complex alarm scheduling, integrating with voice assistants (like Alexa or Google Home), or expanding to include other IoT devices.

Samples

Alarm Clock



08:29:57 PM

Hour



Minute



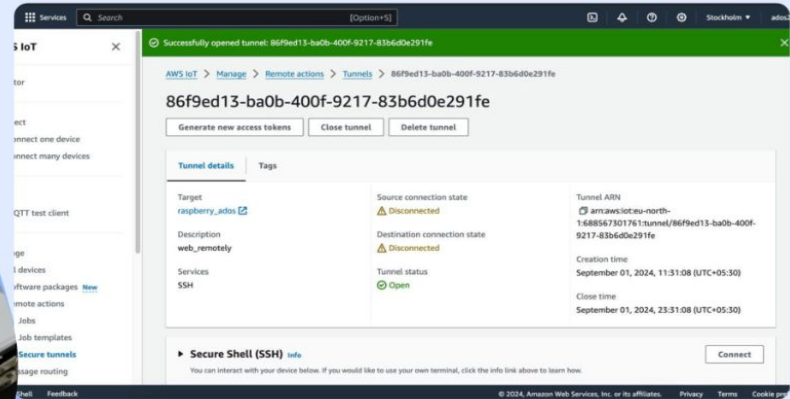
AM/PM



Set Alarm

Previous Data

Snooze



Database

Amazon DynamoDB

A fast and flexible NoSQL database service for any scale

DynamoDB is a fully managed, key-value, and document database that delivers single-digit-millisecond performance at any scale.

Get started

Create a new table to start exploring DynamoDB.

Create table

Pricing

DynamoDB charges for reading, writing, and storing data in your DynamoDB tables, along with any optional features you choose to turn on. DynamoDB has on-demand capacity mode and provisioned capacity mode, and these modes have pricing for processing reads and

How it works



Citations



- [AWS IOT- Device to Dashboard](#)
- [Connecting Raspberry Pi to AWS IOT Core](#)
- [AWS IOT core with Raspberry Pi](#)
- [AWS wiki](#)
- [Node js documentation](#)

