**DESIGN**

**3.1 System Design**

Fig: System Architecture

Sensors

Raspberry Pi Board

AWS IoT

Subscriber

The main components of the system are described below:

* Sensors
* Raspberry Pi Board
* AWS IoT
* Subscriber

**Sensors**

Two HX711 Weight Sensor are used to read the weight of the driver and child which are placed under the seat. These sensors are transducers that converts a load or force acting on it into an electronic signal. These Sensors are connected to Raspberry Pi Board.

**Raspberry Pi Board**

The Raspberry Pi 3 Model B board is used which adds Wireless LAN and Bluetooth Connectivity. It receives the sensor data, performs computing and acts as a connecting bridge between Sensors and AWS IoT.

It sends data to the connected IoT Devices.

**AWS IoT**

The AWS IoT enables Internet-connected devices to connect to the AWS Cloud and lets applications in the cloud interact with Internet-connected devices. Common IoT applications either collect and process telemetry from devices or enable users to control a device remotely. Communication between a device and AWS IoT is protected through the use of X.509 certificates. It receives notifications wirelessly and transmit notifications to the user.

**Subscribers**

Subscribers are the persons who uses the system when need of assistance. They get the notification from AWS IoT if they are the subscribers of the AWS IoT.

3.1.1 Entity Relationship Diagram

Child

Driver

Embedded System

AWS IoT

Sensors

Monitor

Monitor

Communicates with

Alerts

**3.1.2 Data flow diagram**

**3.1.2.1 Level 0 Data Flow Diagram**

Child Alert System

Output data

Input data

Alert

Sensors

**3.1.2.2 Level 1 Data Flow Diagram**

AWS IoT

Send an Alert

Alert

Send an Alert

Embedded System

Sensors

Sensor Status

3.3 Detailed Design

AWS IoT Module

**Device Gateways**

Security and Identity

Device Gateways

Subscribers

Message Broker

Rule Engine

Amazon SNS

Enables devices to securely and efficiently communicate with AWS IoT.

**Message Broker**

The AWS IoT message broker makes it possible for clients to communicate with AWS IoT and for AWS IoT to communicate with clients. Clients send data by publishing a message on a topic. Clients receive messages by subscribing to a topic. When the message broker receives a message, it forwards the message to all clients subscribed to the topic.

Devices connected to AWS IoT are represented by AWS IoT thing in the AWS IoT registry.

An AWS IoT policy is a JSON document that contains one or more policy statements. Each statement contains:

* Effect, which specifies whether the action is allowed or denied.
* Action, which specifies the action the policy is allowing or denying.
* Resource, which specifies the resource or resources on which the action is allowed or denied.

Device Gateway

Policy

Certificates

Thing

Message Broker

Attach to Certificate

Use the Certificates to connect to Thing

Acts as

Interconnected to Certificate

Attach to Certificate

Connect to Thing

Steps:

1. To register the thing, create a single thing. To create a thing, we should add a certificate for thing.
2. One-click certification is selected and activated in which the AWS creates a certificate for thing, private key and a root CA for AWS IoT. Download and save the certificates for further process.
3. Create a Policy, attach the policy and thing to the certificate.
4. Thing acts as Message Broker.

**Rule Engine**

Rules give our devices the ability to interact with AWS services. Rules are analyzed and actions are performed based on the MQTT topic stream. Rules Augment or filter data received from a device and Send a push notification to all users using Amazon SNS.

Message Broker

Amazon SNS

SNS Topic

Role

Rule

Send data

Gives permission to publish

Topic is Created

Attach topic ARN to Rule

Send Filtered data when triggered

**Steps:**

1. Create a rule by selecting the SQL version and modifying the rule query statement according to actions required.
2. Send a message as an SNS push notification action is added to rule by giving the ARN of the created SNS topic.
3. Permission is given to access and publish by creating and updating a role.

**Amazon SNS**

Amazon SNS

Subscribers

SNS Topic

Subscription

Create a Topic

Add Endpoints

Send Data

Add Subscribers

**Steps:**

1. Create an SNS topic, An ARN is created for SNS topic.
2. To receive SMS messages on cell phone, subscribe to the Amazon SNS topic.
3. Subscription is done by enabling the SMS protocol and entering the endpoint which is the cell number.