# Engagement API - Code, Queries, and APIs

## SQL Queries for Table Creation

-- Engagement Post Table  
CREATE TABLE engagement\_post (  
 id SERIAL PRIMARY KEY,  
 tenant\_id INTEGER,  
 post\_type VARCHAR(50),  
 created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP  
);  
  
-- Engagement Post Content Table  
CREATE TABLE engagement\_post\_content (  
 id SERIAL PRIMARY KEY,  
 post\_id INTEGER REFERENCES engagement\_post(id),  
 content\_url TEXT,  
 story\_id INTEGER  
);  
  
-- Engagement Post Product Mapping Table  
CREATE TABLE engagement\_post\_product\_mapping (  
 id SERIAL PRIMARY KEY,  
 post\_id INTEGER REFERENCES engagement\_post(id),  
 product\_id INTEGER  
);  
  
-- Engagement Post Product Table  
CREATE TABLE engagement\_post\_product (  
 id SERIAL PRIMARY KEY,  
 product\_name VARCHAR(255),  
 product\_image TEXT,  
 sku VARCHAR(50)  
);  
  
-- Collection Table  
CREATE TABLE collection (  
 id SERIAL PRIMARY KEY,  
 collection\_name VARCHAR(255)  
);  
  
-- Engagement Post Collection Table  
CREATE TABLE engagement\_post\_collection (  
 id SERIAL PRIMARY KEY,  
 post\_id INTEGER REFERENCES engagement\_post(id),  
 collection\_id INTEGER REFERENCES collection(id),  
 duration\_in\_seconds INTEGER  
);

## API Endpoints

// Fetch Posts for a given tenant\_id  
const pool = require('../config/dbConfig');  
  
exports.getPosts = async (req, res) => {  
 const { tenant\_id } = req.params;  
  
 try {  
 const result = await pool.query(`  
 SELECT p.id, c.content\_url, pr.product\_name  
 FROM engagement\_post p  
 JOIN engagement\_post\_content c ON p.id = c.post\_id  
 JOIN engagement\_post\_product\_mapping pm ON p.id = pm.post\_id  
 JOIN engagement\_post\_product pr ON pm.product\_id = pr.id  
 WHERE p.tenant\_id = $1  
 `, [tenant\_id]);  
  
 res.json(result.rows);  
 } catch (err) {  
 console.error(err);  
 res.status(500).json({ error: 'Something went wrong' });  
 }  
};

// API to Create New Product  
exports.createProduct = async (req, res) => {  
 const { product\_name, product\_image, sku } = req.body;  
 try {  
 const result = await pool.query(`  
 INSERT INTO engagement\_post\_product (product\_name, product\_image, sku)  
 VALUES ($1, $2, $3) RETURNING \*;  
 `, [product\_name, product\_image, sku]);  
  
 res.status(201).json(result.rows[0]);  
 } catch (err) {  
 console.error(err);  
 res.status(500).json({ error: 'Something went wrong' });  
 }  
};