Project Architecture – TechBay

1 High-Level Layers and Responsibilities

1.1 Frontend Layer

- Static Hosting (S3; optional CloudFront): Serves the SPA assets (HTML/CSS/JS).
- Browser SPA: Handles UI, authentication flows (via Cognito), and REST calls to backend API.

1.2 Authentication Layer (Amazon Cognito)

- User Pool: Manages sign-up/sign-in, issues JWTs.
- User Groups: users (regular) vs. admins (elevated).
- Lambda Triggers:
  - Pre-Signup: Validates email domain.
  - Post-Confirmation: Auto-adds new users to users group.

1.3 API Layer (API Gateway)

- REST API with Cognito Authorizer: Validates JWT on each request.
- Resources/Methods:
  - /products (GET),
  - /products/{productId} (GET),
  - /cart (GET, POST, DELETE),
  - /orders (POST, GET),
  - /admin/products (DELETE),
  - /admin/laptops, /admin/pcscreens, /admin/pcs, /admin/headphones, /admin/televisions, /admin/tablets, /admin/phones (POST).
- Integration: Lambda proxy to business-logic Lambdas; CORS enabled for SPA origin.

1.4 Business Logic Layer (AWS Lambda Functions)

- TechBay-AutoAssignGroup: Adds registered users to the users group.
- TechBay-GetProducts-dev: Retrieves products from all product tables (Laptops, PCScreens, PCs, Headphones, Televisions, Tablets, Phones).
- TechBay-AddToCart-dev: Adds a product to the cart, writes to TechBayCart-dev table.
- getProductsFromCart: Retrieves products from the cart by userId and timestamp from TechBayCart-dev table.
- removeProductFromCart: Removes products from the cart by userId from TechBayCart-dev table.
- TechBay-GetProductById-dev: Retrieves specific product data by productId from the relevant product table.
- purchasingProducts: Processes purchase of products in the cart, writes to TechBayOrders-dev table, publishes SNS notification.
- Admin Functions:
  - TechBay-DeleteProduct-dev: Deletes a product by ID from any product table.
  - TechBay-AddLaptop-dev: Adds a new laptop to TechBayLaptops-dev table.
  - TechBay-AddPCScreen-dev: Adds a new PC screen to TechBayPCScreens-dev table.
  - TechBay-AddPCS-dev: Adds a new PC to TechBayPCs-dev table.
  - TechBay-AddHeadphone-dev: Adds new headphones to TechBayHeadphones-dev table.
  - TechBay-AddTelevision-dev: Adds a new television to TechBayTelevisions-dev table.
  - TechBay-AddTablet-dev: Adds a new tablet to TechBayTablets-dev table.
  - TechBay-AddPhone-dev: Adds a new phone to TechBayPhones-dev table.
  - getPurchasesOrders: Retrieves orders from TechBayOrders-dev table.
- Config Access: Reads SNS topic ARN from Parameter Store (cached per invocation).

1.5 Data Layer (DynamoDB)

- Tables:
  - TechBayCart-dev: PK: userId; SK: productId#timestamp. Stores cart items.
  - TechBayOrders-dev: PK: userId; SK: orderId. Stores orders.
  - Product Tables (TechBayLaptops-dev, TechBayPCScreens-dev, TechBayPCs-dev, TechBayHeadphones-dev, TechBayTelevisions-dev, TechBayTablets-dev, TechBayPhones-dev): PK: productId; store product data.

- Attributes: title, description, price, category, createdAt, userEmail (in relevant tables).
- Access: Queries by userId for user operations; scan or secondary index for admin operations.

1.6 Messaging Layer

- SNS Topic: TechBay-Notifications.
- Events: Product purchase (from purchasingProducts).
- Subscriptions: Sends email notifications to admins for orders.

1.7 Configuration & Scheduling

- Parameter Store: Single JSON parameter (e.g., /techbay/config) with SNS ARN. Lambdas read this at startup.

1.8 Monitoring & Audit

- CloudWatch Logs/Metrics: Lambdas log actions; metrics on invocations/errors/durations. Alarms can be set for error spikes.
- CloudTrail: Captures API calls for auditing resource changes and operations.

2 Component Interactions and Data Flows

2.1 Adding a Product to Cart

1. Browser: User selects a product and adds it to the cart.
2. API Gateway: Receives POST /cart with JWT, validates via Cognito authorizer.
3. TechBay-AddToCart-dev Lambda:
    - Extracts userId and email from JWT claims.
    - Reads config (SNS ARN) from Parameter Store (cached).
    - Creates a record with productId and timestamp, writes to TechBayCart-dev table.
    - Returns HTTP 201 with item details.

2.2 Retrieving Products

- GET /products: TechBay-GetProducts-dev aggregates queries from all product tables, returns a list of products.
- GET /products/{productId}: TechBay-GetProductById-dev queries the relevant product table by productId, returns product details.

2.3 Viewing and Managing Cart

- GET /cart: getProductsFromCart queries TechBayCart-dev table by userId, returns cart items.
- DELETE /cart: removeProductFromCart deletes items from TechBayCart-dev table by userId and productId.

2.4 Purchasing Products

1. Browser: User confirms purchase of cart items.
2. API Gateway: Receives POST /orders with JWT.
3. purchasingProducts Lambda:
    - Extracts userId from JWT claims.
    - Reads items from TechBayCart-dev table.
    - Creates an order, writes to TechBayOrders-dev table.
    - Publishes ”order completed” SNS message.
    - Removes items from cart.
    - Returns HTTP 201 with order details.
4. SNS: Sends email notification to admins.

2.5 Admin Operations

- DELETE /admin/products: TechBay-DeleteProduct-dev deletes a product by ID from any product table.
- POST /admin/laptops, /admin/pcscreens, etc.: Functions like TechBay-AddLaptop-dev add products to respective tables.
- GET /orders: getPurchasesOrders retrieves orders from TechBayOrders-dev table for the admin dashboard.

3 Security and Access Control

- Cognito JWT: API Gateway authorizer ensures only authenticated requests reach Lambdas.
- Groups: Lambda logic checks JWT groups claim. Regular users can operate only on their cart and orders; admins have broader access (e.g., manage all products and orders).
- IAM Roles:
    - TechBay-AddToCart-dev, purchasingProducts: DynamoDB (read/write to TechBayCart-dev, TechBayOrders-dev), SNS publish, Parameter Store read.
    - TechBay-GetProducts-dev, TechBay-GetProductById-dev: DynamoDB (read from product tables).

  - removeProductFromCart, getProductsFromCart: DynamoDB (read/delete from TechBayCart-dev).
  - Admin Lambdas: DynamoDB (read/write/delete to product and order tables).
  - TechBay-AutoAssignGroup: cognito-idp:AdminAddUserToGroup.
- Parameter Store: Centralizes configuration (SNS ARN).
- CORS: API Gateway configured to allow only the SPA origin.
- Input Validation: Lambdas validate incoming data (e.g., required fields, price format).
- Least Privilege: Avoid wildcard permissions; scope to resource ARNs.
- CloudTrail & CloudWatch: Track changes and runtime errors.

4 Deployment & Configuration

- Infrastructure as Code: Use CloudFormation (or AWS SAM/CDK) template to create:
  - S3 bucket (static hosting), Cognito User Pool & app client (with redirect URIs), DynamoDB tables, SNS topic & subscription, Lambdas (with code packages), API Gateway resources and authorizer, CloudTrail trail.
- Frontend Update: After deployment, update js/config.js with Cognito domain, clientId, redirectUri, and API endpoint. Upload updated files to S3.
- Lambda Code: Package each function with dependencies. Rely on Parameter Store for external configuration.
- Parameter Store Initialization: Write SNS ARN to /techbay/config after resource creation.
- Testing:
  - Sign up a user, verify email domain validation and group assignment.
  - Add products to cart, complete a purchase, verify SNS notifications.
  - Add/delete products as admin, verify orders.
- Monitoring: Set up CloudWatch dashboards/alarms for Lambda errors; ensure CloudTrail is logging.

5 Simplified Sequence Overview

1. User Sign-Up: Browser → Cognito (with PreSignupValidation and TechBay-AutoAssignGroup Lambdas) → user in users group.
2. User Authenticated: Browser holds JWT.
3. Cart Operations: Browser (JWT) → API Gateway (Cognito authorizer) → TechBay-AddToCart-dev/removeProductFromCart/getProductsFromCart Lambda → DynamoDB (TechBayCart-dev).

4. Purchase: Browser → API Gateway → purchasingProducts Lambda → DynamoDB (TechBayOrders-dev, TechBayCart-dev) → SNS notification → Admin email.
5. Admin Operations: Browser → API Gateway → Admin Lambdas → DynamoDB (product and order tables).
6. Config Flow: Lambdas read SNS config from Parameter Store.
7. Monitoring/Audit: CloudWatch collects logs/metrics; CloudTrail logs all API calls.