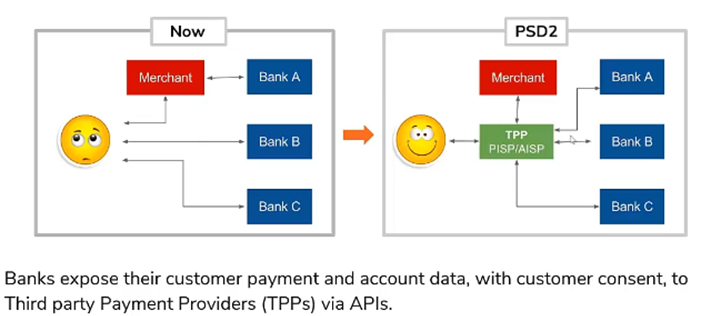
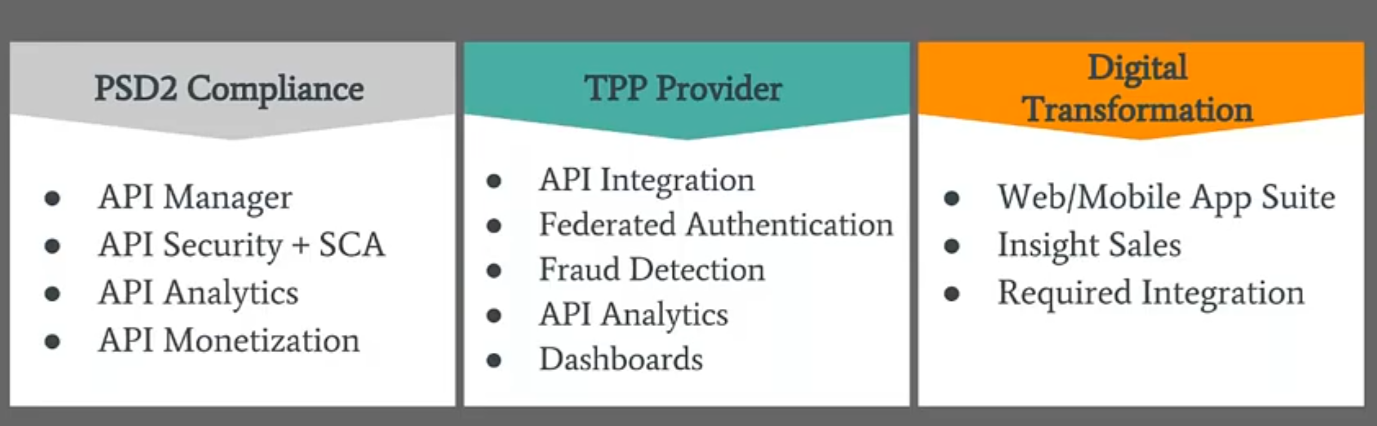
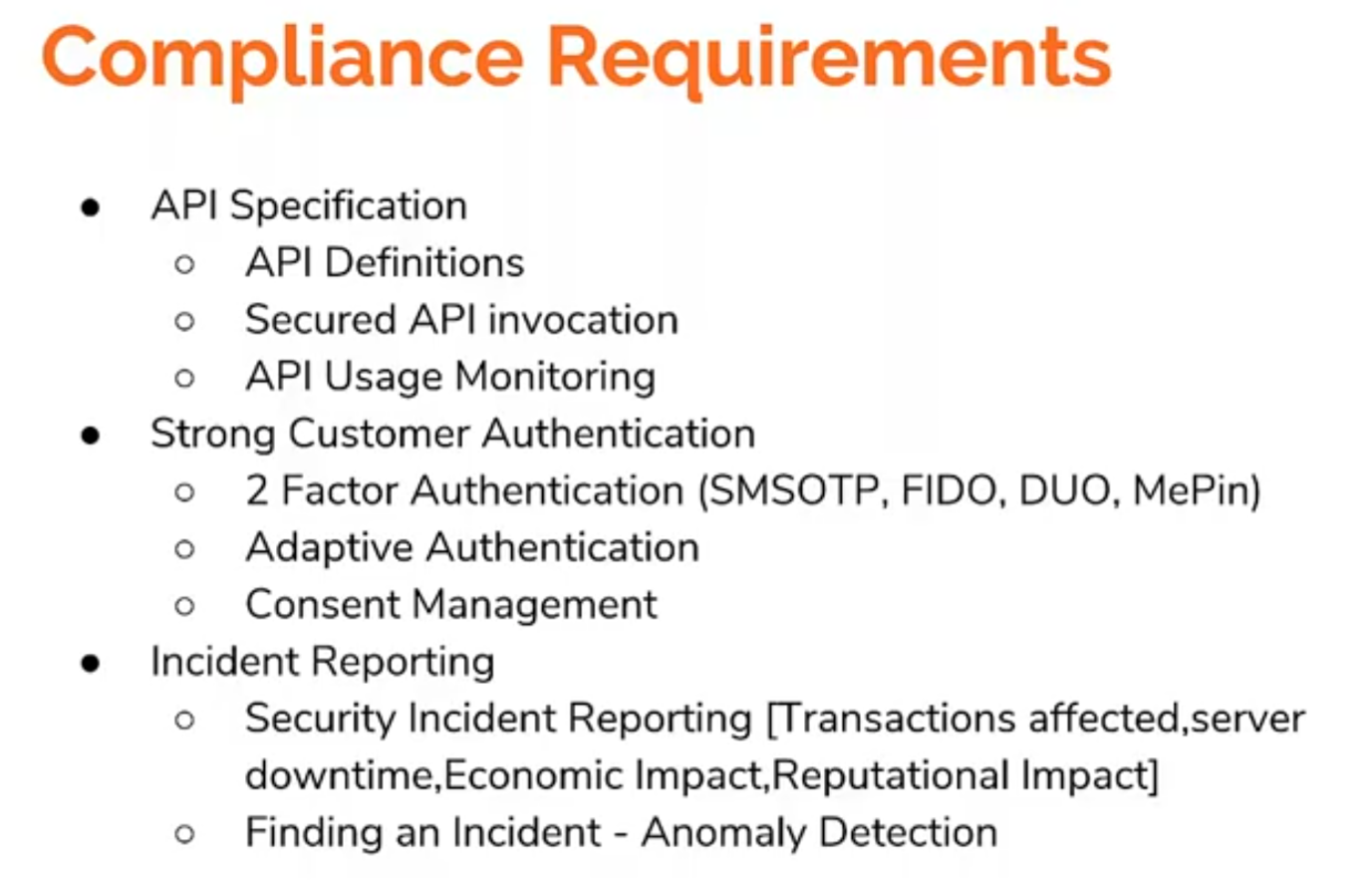
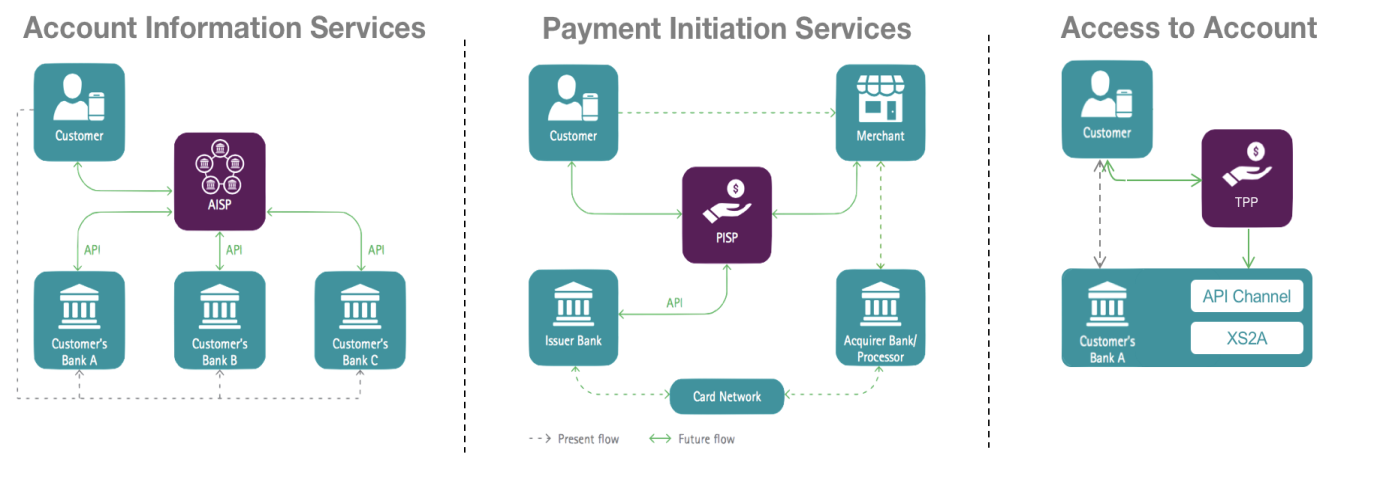
**WHAT IS PSD2**



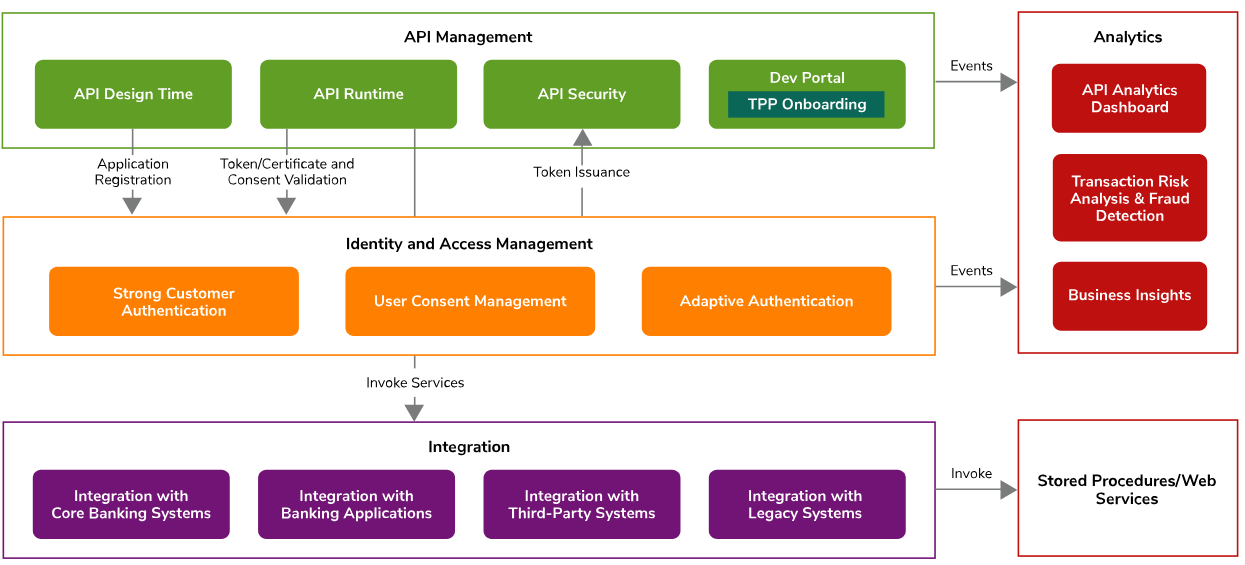


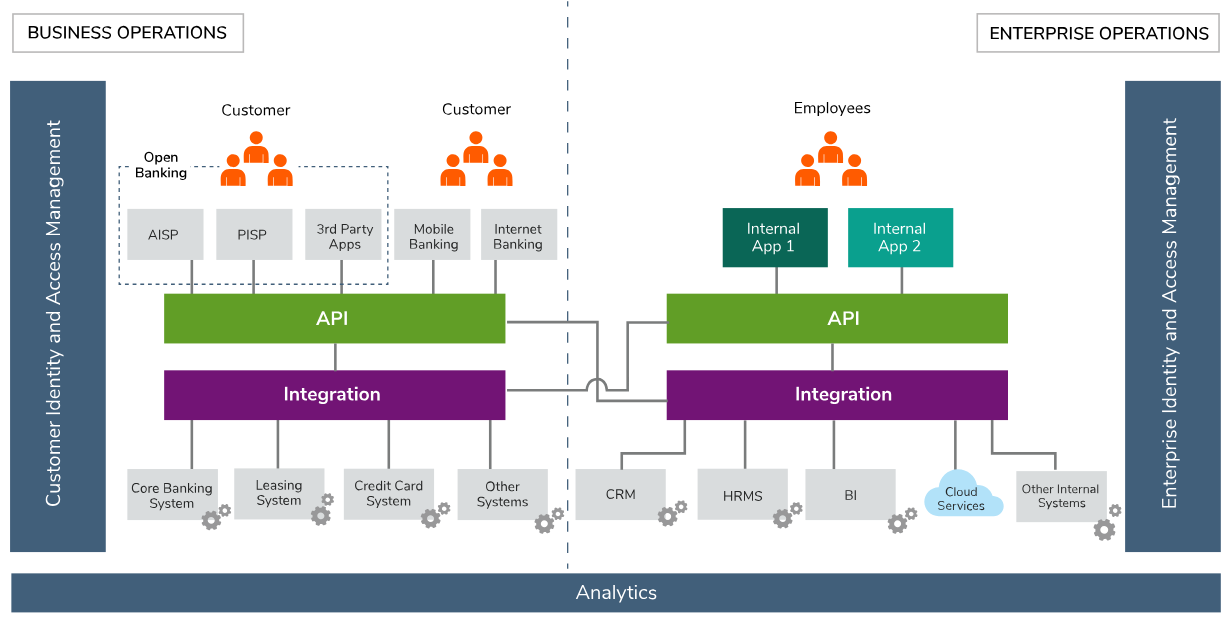


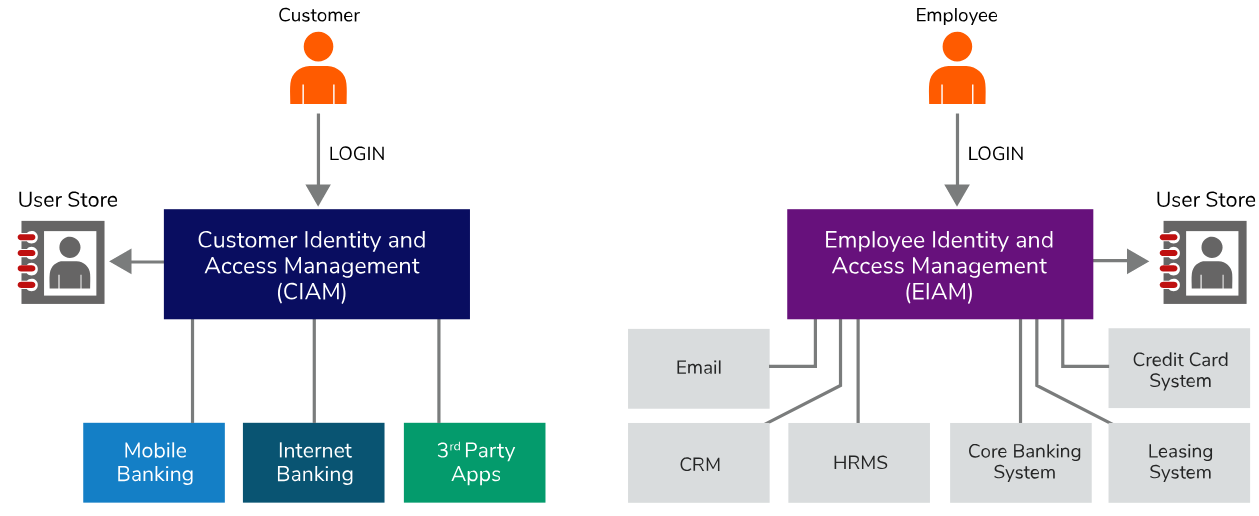


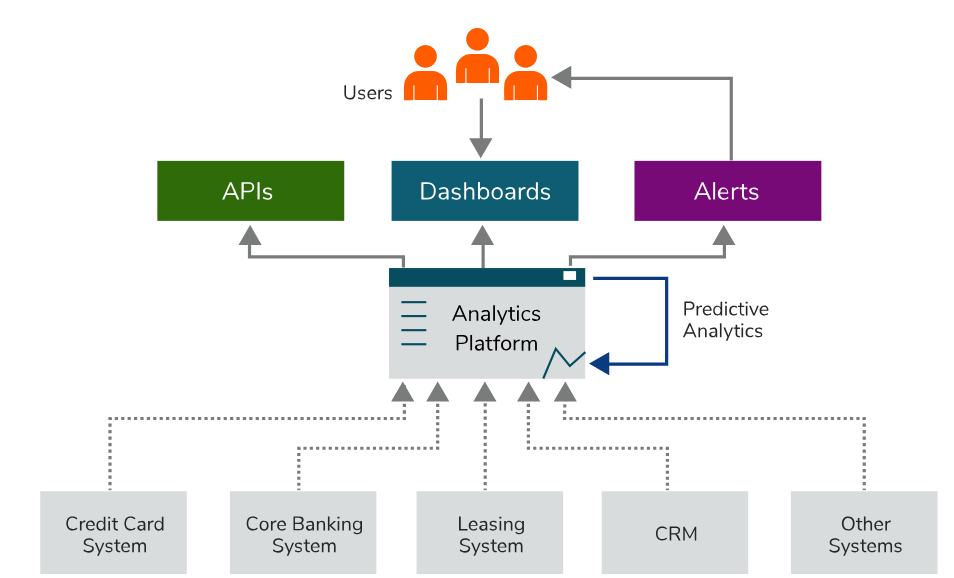


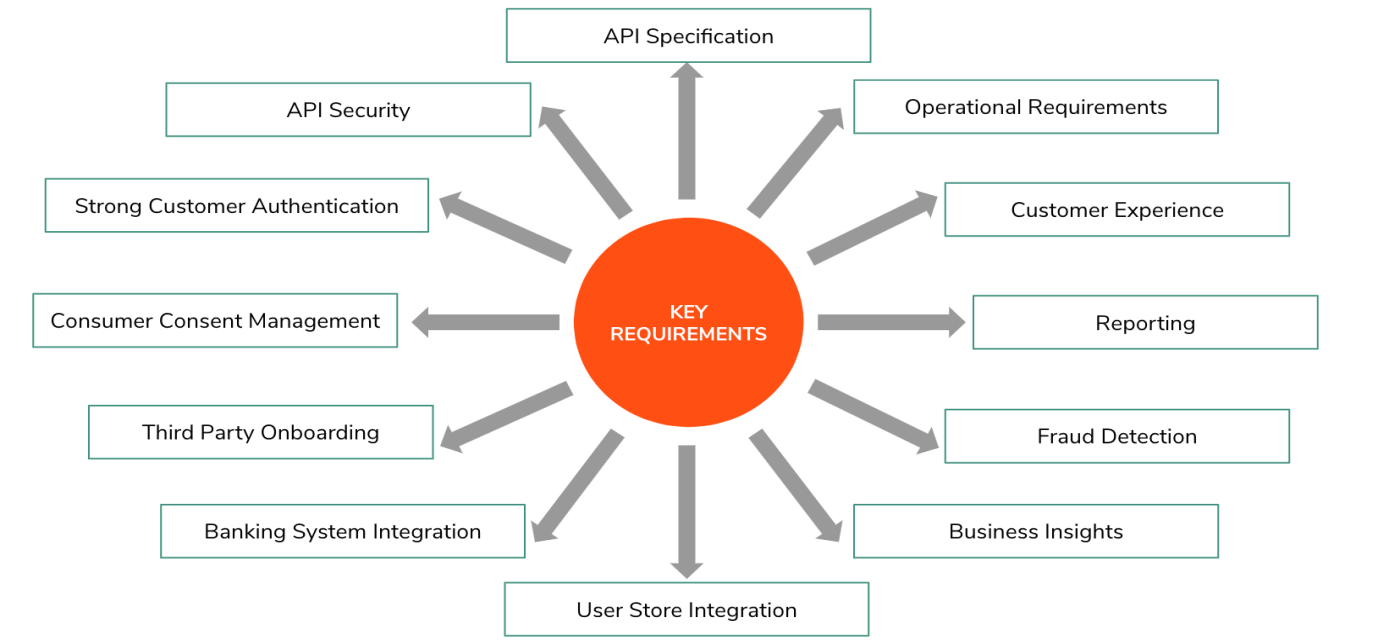
High Level Design











**API Specification**

Firstly, each bank definitely needs to define a proper API specification to ensure how a bank exposes its internal data and services to external parties are standardized and well-defined. When thinking about existing data and services there can be some set of data that can be exposed via **open APIs**. For example, ATM locations, branch locations, exchange rates, and interest rates can be exposed by means of open APIs. But if a bank wants to expose account information of bank customers or needs to provide a service to make payments, those APIs need to expose as **secured APIs.**

**API Security**

Once the API specification is defined, and then exposed to the outside, banks need to think about how to restrict access to the APIs to authorized third parties only. Banks need to implement a security layer for exposed APIs. Mainly **OAuth2 token** or **certificate based** third party authentication and authorization mechanisms are widely used in different open banking systems.

**Strong Customer Authentication**

When sharing customer data with third parties, banks need to get customer consent. In doing that first the bank needs to strongly identify the customer. Authenticating users only via one authentication factor is not enough. There should be multi-factor authentication where at least a combination of two factors of knowledge, ownership, and inherence should be used

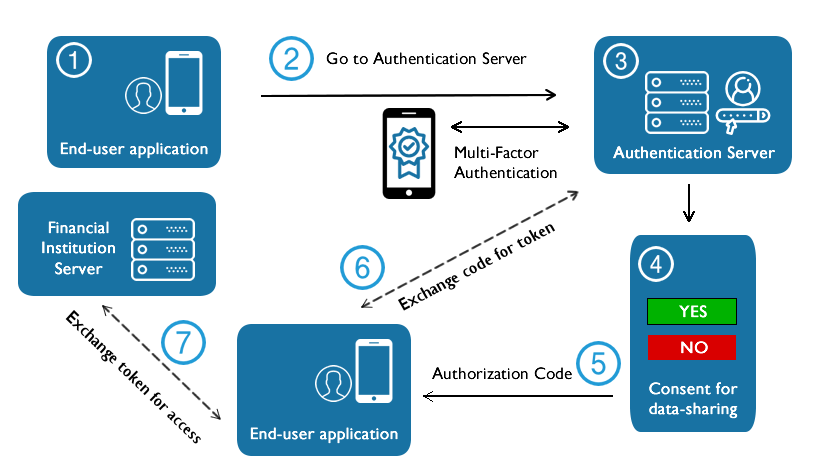
**Authentication factors from basic, SMSOTP, Vasco, fingerprint, voice, Facebook, Google, and more**

**Consumer Consent Management**

Managing consent means it gives an authority to the bank customer to control his personal and financial data in terms of whom they may be shared with, for what purpose and for what period. The open banking platform should have the capability to capture, store and validate this consent when sharing customer data with third parties.

* **Service Authentication (Bank)**

Depends on bank -> Redirect to bank aap or authentication through phone link to Bank



* **Consent Revocation**

1. Provides an interface for bank users to log in and revoke the consents.
2. Provide an interface to customer care officers to search for and revoke the consent on behalf of the customer
3. Provide an API to revoke the consent so that third parties can provide a revoking functionality through their applications.

**Third-Party On boarding, management and monitoring**

Process to integrate a Bank or any other third party service.

When exposing an API, we need to connect to our existing banking systems. Those banking systems may work with **different message formats (JSON, XML)**and **different massage transports (HTTPS/S, VFS, JMS, TCP)**. So the open banking platform should have the capability to connect with any type of internal or external banking system.

**User Store Integration**

* Adoption of Security standards and applicable guidelines such GDPR.
* Way to storage **LDAP, AD or JDBC.**

**Business Insights, Fraud Detection, and Reporting**

* When considering the data that passes through an open banking platform, we can see spending patterns of bank customers and identify some **business insights** to improve the banking business.
* Especially when making payments through the open banking platform, frauds can happen. So there should be proper **fraud detection** solution connected to this platform and if the bank already has a fraud detection solution a bank can be able to connect it without buying new solution.
* **Reporting** capabilities are needed to generate reports for bank management, third parties and for relevant stakeholders to see how the open banking platform is performing and to take necessary business decisions.
* **Transaction Risk Analysis** There should be a capability to identify the risk level of a transaction and if it is low then the bank can exempt the user from having to go through all the authentication factors. That is what we call [Transaction Risk Analysis (TRA)](https://wso2.com/articles/2019/05/a-deep-dive-of-transaction-risk-analysis-for-open-banking-and-psd2). For example if transaction is more than 1000 GBP add required SMS authentication.

**Customer Experience**

* Have a simple and easy navigation without any delay.
* Easy as how a bank customer would communicate with the bank directly
* Required and correct information for bank customers to take a decision
* User interfaces, emails, alerts, reports and error messages

**Operational Requirements**

* The bank should properly design how to test **and verify the whole platform before putting it** in production
* Customer care portal
* Third-party providers (**TPPs**) need to be able to rely on highly available and well-performing dedicated interfaces provided by Account Servicing Payment Service Provider (**ASPSPs**), so that they can, in turn, provide reliable services to their customers.
* Provide **testing facilities to the** third parties before their application goes live so that banks can identify and fix issues early.
* **Audit Trail** All traffic passing through an API gateway and the application stack can be tracked and logged to ensure visibility and in such a way that the logs cannot be tampered with. Each application service logs all events to a dedicated audit service, with the audit trail analysed in the Analytics Tier.