Open Banking Technology Framework

Central Bank of Oman

11 01 2024

Table of Contents

1 Introduction to Open Banking Technology Framework

Open Banking Technology Framework is a comprehensive framework that entails a blueprint for enabling secure sharing of financial data and facilitates the development of innovative financial services and applications. It is a set of principles, standards, and protocols that govern the way financial institutions and third-party providers interact and exchange data. At its core, the Open Banking Technology Framework promotes transparency, customer control, and competition in the financial sector. It allows customers to securely share their financial information with authorized third-party providers, such as fintech companies, to access a wide range of services and products. This framework relies on the use of open application programming interfaces (APIs) that facilitate seamless data sharing between different systems and platforms.

The key components of the Open Banking Technology Framework include strong data security measures, standardized APIs, consent management mechanisms, and authentication protocols. These elements ensure that customer data is protected and accessed only with explicit consent from the customer. Additionally, the framework promotes interoperability, enabling different financial institutions and service providers to collaborate and offer integrated services to customers. Open banking system provides a user with a network of financial institutions' data using application programming interfaces (APIs).

A Technology Framework for open banking refers to the underlying architecture, infrastructure, and technological components that enable the implementation and operation of open banking systems. It provides a blueprint for designing and developing the necessary technological solutions to facilitate secure data sharing, seamless integration, and enhanced customer experiences in the open banking ecosystem.

- **API Management Platform:** API management platform enable the integration through APIs between different software systems, allowing them to communicate with other Microservices Architecture: It is used for developing software systems that are organized around business capabilities and are independently deployable by fully automated deployment machinery. This architecture helps banks to be more agile and scalable.

- **Consent Management Platform:** The consent management platform is a solution which is used to collect and manage user consents. The solution should be configured to the Oman’s data protection laws and regulations.

- **Identity and Access Management (IAM):** IAM module helps to ensure that only authorized individuals can access certain data. This is crucial for maintaining the security and privacy of user data in an open banking context. The consent management platform should be tightly integrated with the IAM system.

- **Testing and Monitoring Tools:** These tools help ensure that APIs and other open banking systems are working correctly and efficiently. They can identify any issues or bottlenecks that need to be addressed. The testing and monitoring tools should be deployed, owned and managed by both TPPs and the banks for their respective testing.

2 Scope and Objective of this document

The objective of this document is to provide overview of the technology architecture of the open banking and underlying systems. It also tries to cover two important platforms, in detail, that enable open banking along with the other technologies, API management platform and Consent management platform.

It also provides glimpse of various monetization framework that the API producers can use for monetizing their APIs. Others include Key performance indicators (KPIs) for the Open banking as a whole and tools that are prescribed for enabling the open banking in Oman.

What this document is not is,

- The detailed open banking architecture with various integration touch points.
- A manual for solution architects for designing and integrating the systems.
- All the tools are prescribed based on the best practices; they are just examples.

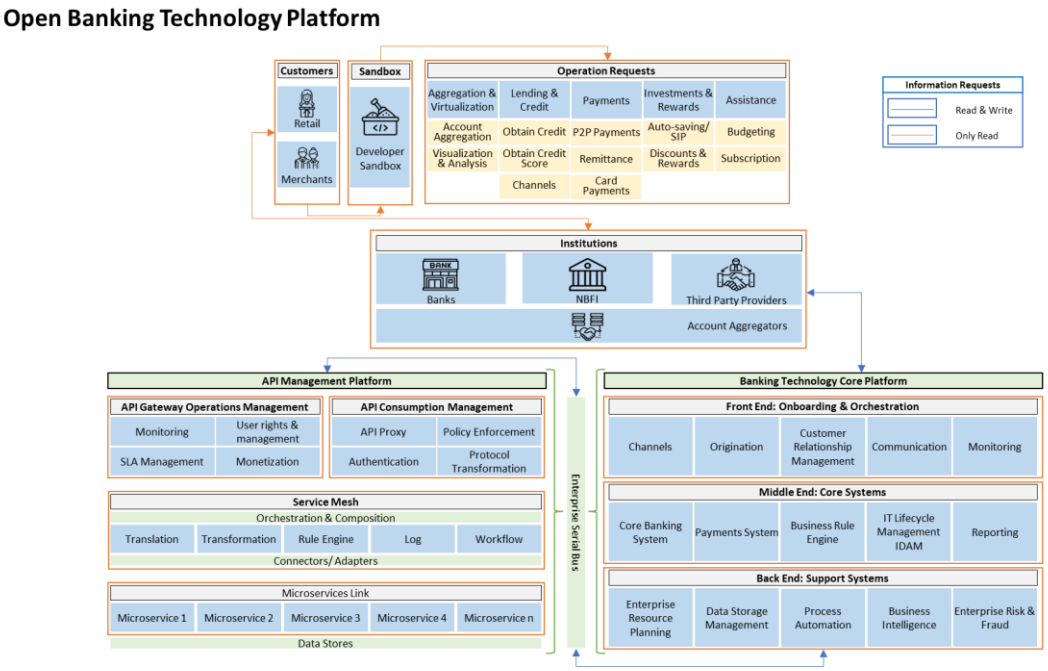
It is prescribed to undergo a thorough assessment of your existing technology stack and design the target state architecture for open banking before implementation of the same.

Further, these specifications should be read in conjunction with the all the existing guidelines and mandates published by the Central Bank of Oman (CBO). All ecosystem participants must ensure compliance to all the relevant existing standards applicable in Oman

3 Components of Technology Framework in Open Banking

The open banking architecture has various components which should be tightly integrated for seamless flow of information. Participants must do a thorough assessment of their existing capabilities in terms of integration, data exposure, product linkages etc. prior to finalizing a vendor or implementing the Open banking.

.



Various components of an Open Banking Framework include,

3.1 Third party providers (TPPs)

Third Party Providers are organisations or natural persons that use APIs developed to Standards to access customer's accounts, to provide account information services and/or to initiate payments. Third Party Providers are either/both Payment Initiation Service Providers (PISPs) and/or Account Information Service Providers (AISPs).

The AISPs are entities that are authorized to access financial information from different banks and financial institutions with the user's consent where as PISPs are entities that facilitate online payments on behalf of consumers or businesses by initiating transactions directly from their bank accounts. These transactions are often initiated through open banking APIs (Application Programming Interfaces) with the user's consent.

The TPPs should be onboarded with the bank before they begin integration and offering products and services to the end customers. For more information, please refer to Licensing documentation.

The customer facing applications are the ones that are provided by the Third-Party Service providers (TPPs) to the end customers. These include web applications, mobile applications etc. While

designing the end user applications, the TPPs should consider design principles outlined in section 3.1.1 to provide best customer experience.

3.1.1 UI/UX Design principles

UI/UX is paramount for the success of Open Banking as it directly impacts user trust, engagement, and adoption. A well-designed and intuitive user interface ensures that customers can easily navigate through financial services, view their data securely, and perform transactions seamlessly. This not only enhances the overall user experience but also fosters confidence in the platform's security and reliability, ultimately driving greater adoption and satisfaction among users, which is crucial for the success and widespread acceptance of Open Banking initiatives. The UI/UX principles should be predominantly followed by the TPPs who provide services to the end customers.

UI/UX design is a critical element in the success of Open Banking, influencing user trust, adoption, data security, customer satisfaction, and overall competitiveness in the financial industry.
By adhering to the following UI/UX principles, Open Banking applications can offer users a highly effective, secure, and enjoyable digital banking experience:

- **User-Centered Design:** At the core of effective UI/UX design in Open Banking is the principle of being user-centered. This involves conducting thorough research to gain a deep understanding of the users' needs, preferences, behaviors, and goals. By putting the user at the center of the design process, tailor made interfaces can be created to meet their specific requirements, ultimately leading to a more satisfying and efficient user experience.

- **Consistency:** Consistency in UI/UX design refers to the uniformity of elements and interactions throughout the application. This means maintaining a consistent visual language, including elements such as typography, colors, buttons, and icons. When users encounter familiar elements and patterns consistently, they feel more at ease and are better able to navigate the application intuitively.

- **Simplicity:** Simplicity is about distilling the interface to its essential elements while eliminating unnecessary complexities. A simple interface reduces cognitive load for users, making it easier for them to understand how to use the application and access the information or services they need. Clutter-free design is key to achieving simplicity. Clarity is closely related to simplicity and involves ensuring that all elements of the interface are clear and easily understandable. This includes using concise and jargon-free language, providing clear labels for buttons and menu items, and using intuitive icons and symbols. Ambiguity and confusion should be minimized to enhance the user experience.

- **Hierarchy:** Establishing a clear visual hierarchy in the design is essential for guiding users' attention. Key information and primary actions should be prominently displayed, while secondary or less important elements should be less emphasized. This hierarchy helps users quickly identify what is most relevant to their current task.

- **Accessibility:** Accessibility in UI/UX design is about making the application usable by individuals with disabilities. This involves adhering to accessibility standards outlined by

Central bank of Oman. Few examples include providing alternative text for images, ensuring keyboard navigation is possible, and offering features like screen reader compatibility. Accessibility considerations ensure inclusivity in the user base.

- **Feedback:** Providing timely and informative feedback to users is crucial for a responsive and user-friendly interface. Users should receive feedback in the form of visual or auditory cues when they perform actions, such as button clicks or form submissions. This feedback helps users understand that their actions have been registered and what to expect next.

- **Mobile Responsiveness:** In the era of multiple devices and screen sizes, mobile responsiveness is paramount. A well-designed interface should adapt seamlessly to various screen sizes, ensuring a consistent and user-friendly experience whether accessed on smartphones, tablets, or desktop computers.

- **Loading Speed:** Page loading speed can significantly impact the user experience. Slow-loading pages can lead to user frustration and abandonment. Therefore, optimizing loading times through techniques like image compression and efficient coding is vital to maintaining a positive user experience.

- **Cyber Security:** Cyber Security is of utmost importance in Open Banking, and the UI/UX design should reflect this. Clearly communicate security measures to users, using familiar symbols and terminology. Provide assurances regarding data protection and privacy to build trust and confidence in using the application.

- **User Testing:** User testing involves real users interacting with the interface to identify issues and gather feedback for improvements. Regularly conducting user tests helps in refining the design based on actual user experiences, ensuring that it aligns with user expectations and needs.

- **Aesthetics:** While functionality is primary, aesthetics should not be overlooked. A visually appealing interface can enhance the overall user experience by creating a positive impression and making users more inclined to engage with the application.

- **Scalability:** Design interfaces with future growth in mind. Anticipate the addition of new features and services without causing disruptions to the user experience. Scalable design ensures that the application can evolve seamlessly over time.

- **Error Handling:** Plan for and design error messages and recovery processes. When errors occur, users should receive clear and informative error messages that help them understand what went wrong and provide guidance on how to correct the issue.

- **Iterative Design:** UI/UX design is an ongoing, iterative process. Continuously gather user feedback and analyze user behaviors to identify areas for improvement. Regularly update and refine the design to ensure it remains aligned with changing user needs and technological advancements.

Since open banking requires explicit consent to be given for sharing their data, customer experience becomes a crucial aspect. The customer should be,

- Properly informed about the intention of consent acquisition. The consent should be clear and understandable by the customer and must clearly articulate the purpose of the consent.
- Feel secure while sharing the data to the third-party providers.
- Have complete control of the consent process including purging the customer data as per the CBO guidelines.

Banks should take a balanced approach while defining the customer experience journey in the open banking. The approach should satisfy two stakeholders who are responsible for designing and building the experience, the regulator, and the developers. The guidelines from the regulators should serve as the roadmap from adherence to the regulation perspective, while there should also be flexibility for the developers to experiment with the customer experience in terms of UI/UX and the journey.

Banks can also use the UI/UX guidelines as additional layer of security for the customers. The bank can build functionality which restricts 'screen-scraping'. The UI/UX guidelines can also be used to develop a more engaging process for data collection and sharing.

Some of the best practices in developing innovative customer experience:

- The data to be shared should be visible to the customer before consenting to share
- The UI/UX should be practical and minimal. This helps restrict project of unnecessary information to the customer.
- Provision of functionalities through omni-channel.
- The experience should be evolving based on the customer interaction and feedback. This means the underlying technology should be agile and allow the modifications without much hassle.
- All the documentation related to customer experience should be proper and updated frequently.

The experience should be seamless throughout the journey. Payment (open banking) use case can be effective if the bank created value at each touchpoint of the journey. The examples include,

- Alternative payment methods
- Payment automation
- Real time settlement

Codification of agreements (UK, n.d.): The Customer Experience Guidelines (CEG) specifies the format in which the agreements should be developed and displayed to the customer. It should be in the below format,

Agreement Parameter	Description
Customer Outcome Statement	What we aim to help you achieve by using this product or service

Data Usage Statements	This is how we will use (and limit the use of) your data
Managing Your Data Statement	This is how you can manage your data and revoke your consent if you no longer wish to use the service (N.B. This would be achieved through settings or a "dashboard"). Please ensure that any consequences of revocation are made clear to the customer.
Business Monetization Statement	This is how we make money
Complaints Handling Process Statement	Here's how you can get help
Processing Legal Basis Statement	This is the legal basis we rely upon to lawfully process your data (Likely to be Legitimate Interest or Performance of Contract)
Accountability Statement	Here's how we are regulated and how you can find out about the relevant regulation.

Codification of consents

Consent Parameter	Description
Purpose	Why we need you to share your data
Benefit	What you will get from us in return
Data request	What we need you to share
Timeframe	How long we will need access (unless you revoke access)
Agreement statement	Do you agree to share your data with us on the terms above?

3.2 Financial Institutions & Core systems

The financial institutions (FIs) also known as ASPSPs i.e. Account Servicing Payment Service Provider. ASPSPs are entities that provide and maintain payment accounts for customers. These accounts are often accessed by third-party providers (TPPs), such as AISPs (Account Information Service Providers) and PISPs (Payment Initiation Service Providers), through open banking APIs (Application Programming Interfaces) with the customer's consent. They are API producers and distributors through which they provide access to their customer's data residing in their core systems. The FIs should follow the API development and management as described on the API standards and specification document.

3.3 API Management Platform

API Management platform plays a pivotal role in facilitating Open Banking and supporting various Open Banking use cases. With its comprehensive modules and functionalities, it enables financial institutions to embrace the principles of Open Banking and leverage APIs to enhance customer experiences and foster innovation. Let's explore how the different modules of an API Management platform contribute to Open Banking:

- **API Gateway Operations Management:** The API Gateway module ensures secure and efficient communication between banks and third-party providers (AISP/PISPs). It enforces authentication, authorization, and traffic management policies, protecting sensitive data and preventing unauthorized access. It enables banks to expose APIs securely, allowing AISP/PISPs to access customer account information, initiate payments, and provide value-added services.

- **API Consumption Management:** This module empowers banks to establish developer portals where AISP/PISPs can discover and consume available APIs. It provides self-service capabilities for AISP/PISP onboarding, API documentation, and access token management. Banks can foster collaboration with AISP/PISPs, promote innovation, and enable the development of new financial products and services.

- **API Service Mesh:** The API Service Mesh module enables banks to orchestrate and compose APIs, creating composite services that aggregate functionalities from multiple sources. This empowers AISP/PISPs to access and integrate various banking services seamlessly. API adapters facilitate the integration of diverse backend systems, ensuring interoperability between APIs and underlying services.

- **Microservices Link:** The integration between the API Management platform and microservices architecture supports the development of modular and scalable banking services. It enables banks to expose microservices as APIs, facilitating their integration with AISP/PISPs. This agility promotes innovation, as banks can rapidly deploy new services and update existing ones, meeting the evolving needs of customers and AISP/PISPs.

- **Data Stores:** API Management platforms include data stores that capture and analyse API-related metadata, usage statistics, and performance metrics. These insights help banks monitor API usage, identify trends, and optimize services. Data stores enable banks to comply with Open Banking regulations by tracking API usage, consent management, and data sharing activities.

By harnessing the capabilities of an API Management platform, banks can embrace Open Banking and create a thriving ecosystem of collaboration and innovation. They can provide AISP/PISPs with secure access to customer data and services, while maintaining control and ensuring regulatory compliance. Ultimately, an API Management platform facilitates Open Banking by enabling banks to leverage APIs effectively, drive digital transformation, and deliver enhanced financial services to customers.

3.3.1 Integration of Banking Architecture with API Management Platform

The connector between API platforms and banking architecture is typically a middleware or integration layer. This layer acts as a bridge, facilitating communication and data exchange between the API platform and the various components of the banking architecture.

The middleware or integration layer plays a crucial role in enabling seamless integration and interoperability between different systems, applications, and protocols. It ensures that data flows smoothly between the API platform and the core banking systems, databases, payment gateways, and other relevant components of the banking architecture.

The Enterprise Service Bus (ESB)/ connector layer performs several key functions:

- **Data Transformation:** It handles the mapping and transformation of data between the API platform and the banking architecture. This ensures compatibility and consistency of data formats, structures, and protocols.

- **Message Routing:** The connector layer routes messages and requests from the API platform to the appropriate components in the banking architecture. It directs incoming API calls to the corresponding services or functions within the banking system.

- **Protocol Translation:** It handles the translation of different communication protocols between the API platform and the banking architecture. For example, it may translate RESTful API calls from the platform into SOAP or XML messages that the banking system can process.

- **Security and Authentication:** The connector layer enforces security measures, including authentication and authorization, to ensure secure access to banking data and services. It validates the credentials of API consumers, authorizes access to specific resources, and enforces data privacy and compliance regulations.

- **Event Monitoring and Logging**: The connector layer captures and logs events, transactions, and errors for monitoring, auditing, and troubleshooting purposes. It provides visibility into the flow of data and interactions between the API platform and the banking architecture.

By serving as the connector between the API platform and the banking architecture, the middleware or integration layer enables seamless integration, efficient data exchange, and secure communication. It plays a vital role in ensuring the smooth operation of open banking initiatives, enabling financial institutions to leverage APIs, innovate with third-party developers, and deliver enhanced services to customers.

3.4 Consent Management Platform

Consent must be freely given, specific, informed, and unambiguous. This requires that an individual's consent must be given voluntarily without any pressure or influence that could affect his or her choice.

In accordance with the data privacy laws of Oman, consent by the customers should be mandatory for all the use cases. This is to prevent the unauthorised use of customers' personal data by the participants of the open banking. This calls for having a robust consent management system (CMP) for processing customer's data.

The CMP is a software solution that helps the participants to collect and manage consents by the customers. It should allow collect, track, monitor and respond to the data requests and preferences to the consents. The CMP should have the functionalities of Consent collection, Consent management and data processing.

Consent collection: This module should be integrated with all the channels from where the customer consents are being collected i.e. websites, mobile apps, and other digital platforms.

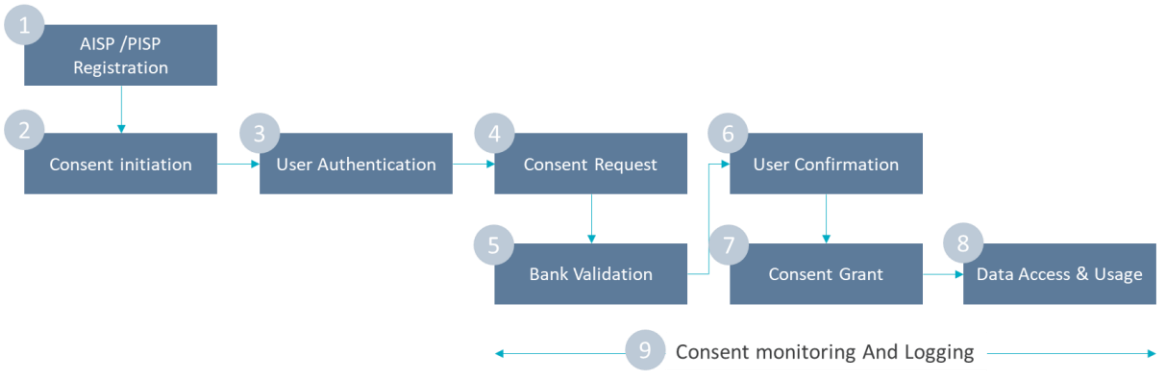
Consent management: Here the data should be harmonised, and a single source of truth should be established for all the consents provided by a customer. The CMP should manage the consent lifecycle.

When a TPP is collecting the consent, the customer should be made aware of the following,

1. Identity of the TPP
2. Purpose of the consent
3. Consent validity

The CMP should also be able to manage one-time consent and recurring consents.

All the participants should be aware and follow the consent management lifecycle.



1. AISP/PISP Registration

The registration of AISP/PISPs is a meticulous process governed by regulatory authorities. AISP/PISPs are required to undergo thorough scrutiny to ensure their legitimacy and compliance with Open Banking regulations. This includes rigorous checks to verify their identity, financial stability, and adherence to security and data protection standards. Authorization to operate as a AISP/PISP is only granted upon successful completion of this process, ensuring that only reputable entities gain access to sensitive financial data.

2. User Initiates Consent

The initiation of consent begins when a user, either an individual or a business entity, recognizes a need to share their financial data with a third-party service provider (AISP/PISP). This decision often arises from a specific financial goal, such as optimizing financial management or utilizing specialized financial services. It's important to note that this step is driven entirely by the user's intention and is at the core of Open Banking's principles of data control and user empowerment.

3. User Authentication

User authentication is the critical gateway to the consent process. Depending on the security measures implemented by the user's bank and the AISP/PISP, this step can involve advanced authentication methods like multi-factor authentication (MFA), biometric verification, or robust

username-password mechanisms. The utmost care is taken to confirm the user's identity securely, as this authentication establishes the foundation for the trustworthiness of the entire consent process.

4. Consent Request

After successful authentication, the AISP/PISP sends a consent request to the user's bank. This request should be meticulously crafted and includes details such as:

- The specific financial data being requested.
- The precise purpose for which the data is needed (e.g., account aggregation, payment initiation).
- The duration for which consent is sought.

The granularity of this request ensures that users have complete transparency about how their data will be used and allows them to make informed decisions regarding access.

5. Bank Validation

The user's bank plays a critical role in ensuring the legitimacy of consent requests. During the validation process, the bank should rigorously check several key aspects, including:

- Verifying the legitimacy and authorization status of the AISP/PISP.
- Conducting a funds check (if applicable) to ensure the user has sufficient funds for the requested transaction.

This scrutiny helps protect users from unauthorized access and ensures compliance with regulatory requirements.

6. User Confirmation

Following successful bank validation, the user is prompted to confirm or deny the consent request. This confirmation is solicited through secure channels, such as a dedicated mobile app, SMS, or email. The user must review and assess the following:

- The specific details of the consent request.
- The implications and potential risks associated with granting consent.

This stage is designed to empower users by giving them full control over their data sharing choices.

7. Consent Grant

If the user decides to grant consent, the bank issues a consent token. This token is a cryptographic key that is:

- Time-limited, specifying a start and end date.

- Uniquely tied to the consent request, ensuring it can't be used for other purposes.
- Protected by robust encryption to prevent tampering or unauthorized access

8. Data Access

Armed with the consent token, the AISP/PISP can securely access the user's financial data through the bank's designated APIs. This access is precise and limited, allowing the AISP/PISP to retrieve only the authorized data for the specified duration. The token acts as an impenetrable key, granting access to the data while maintaining the user's control and privacy.
The AISP/PISP can now employ the accessed data for the precise purpose outlined in the consent request. Whether it involves presenting consolidated account information, initiating payments, or delivering other financial services, the data's usage is strictly bound by the user's consent, ensuring that it is employed only for the intended and authorized activities.

9. Consent Monitoring

Consent monitoring is a continuous process mandated by regulatory authorities. It encompasses the meticulous oversight of consent procedures and data access activities to ensure unwavering compliance with stringent privacy and security standards. This monitoring serves as a protective measure, safeguarding user data and upholding the principles of transparency and accountability.

10. Consent Revoke

The participants should also allow the users to revoke the consent. The Users retain full autonomy over their data throughout the consent management process, including the right to revoke consent at any moment. The mechanisms for revocation are user-friendly, allowing individuals to withdraw their consent either through the bank's interface or by directly communicating with the AISP/PISP. When consent is revoked, it is immediately invalidated, ensuring that the AISP/PISP no longer has access to the user's data.

Audit Trails

Extensive audit trails are meticulously maintained to document and track every aspect of consent-related activities. This audit trail serves multiple purposes, including:

- Enabling transparency for users regarding who accessed their data and when.
- Facilitating compliance checks and audits by regulatory authorities.
- Providing a detailed historical record of consent-related events for dispute resolution if necessary.

Some of the best practices to be considered for consent management are,

- The bank should create a framework for consent management based on the criteria like Who are consent granted, purpose and time.
- There should be an integrated consent management system for managing all requests for the bank.
- The UI / UX enabled to the customer and the AISP/PISP should be same.

Ideally there should be consent APIs developed for each request since the data requested is different for different requests.

3.5 Internal Testing Environment

The financial institutions that are providing access to their customer's data should also facilitate internal testing environment for the TPPs to test their applications and use cases.

3.6 IT Infrastructure requirements for Open Banking

Implementing an open banking mandate requires a robust IT infrastructure and hardware to support the secure sharing of financial data and services. Some of the key components of IT infrastructure and hardware required for an open banking mandate are:

- **Data Centres:** Data centres are the foundation of an open banking infrastructure. Banks need secure, reliable, and scalable data centres to store and process vast amounts of customer data. These data centres should have advanced security measures, redundant power and cooling systems, and disaster recovery capabilities to ensure uninterrupted operation.
- **Servers:** Powerful servers are essential for processing the large volume of data generated by open banking transactions. Banks require high-performance servers to handle API requests, data processing, and real-time analytics. Virtualization technologies can optimize server utilization and improve resource allocation.
- **Network Infrastructure:** A robust and high-speed network infrastructure is critical for the secure transmission of data between different systems and stakeholders. Banks need resilient and secure networks with adequate bandwidth to handle the increased traffic resulting from open banking initiatives. Implementing firewalls, intrusion detection systems, and encryption protocols ensures network security.
- **Storage Systems:** Open banking mandates generate a significant amount of data that needs to be stored securely. Banks require scalable and resilient storage systems, such as network-attached storage (NAS) or storage area networks (SANs), to accommodate the growing volume of customer data. Implementing backup and replication solutions is essential for data protection and disaster recovery.
- **Security Infrastructure:** Given the sensitive nature of financial data, robust security infrastructure is crucial. This includes hardware components such as firewalls, intrusion prevention systems, and secure access gateways. Hardware security modules (HSMs) play a

vital role in securing cryptographic keys and ensuring the integrity of digital signatures and encryption.

- **APIs and API Gateways:** API infrastructure is a core component of open banking. Banks need to implement API gateways, which serve as the entry point for API requests, manage authentication and authorization, and enforce security policies. API management platforms help in API lifecycle management, developer onboarding, and analytics.

- **Endpoint Devices:** Open banking mandates involve interactions with customers through various endpoint devices such as smartphones, tablets, and computers. Banks need to ensure compatibility and security across different platforms and operating systems to provide a seamless and secure customer experience.

To meet the infrastructure and hardware requirements for an open banking mandate, banks must invest in robust and scalable IT infrastructure, implement advanced security measures, and ensure seamless integration between different systems. By having a reliable and secure IT infrastructure, banks can successfully implement open banking initiatives, enhance customer experiences, and foster innovation in the financial industry

4 Roles and responsibilities of the participants

4.1 Financial Institutions: Banks provide access to customer's data to TPPs through secure APIs, based on customer's request. There are certain requirements for Open banking compliance,

- Require strong internal governance and technical expertise to efficiently coordinate and aggregate multiple and disparate data sources
- Must maintain records of consent for audit purposes, vet partners and their cybersecurity capability.

4.2 Third Party Providers (TPPs): They leverage technology and data available via Open Banking to develop solutions for banks that can be integrated with banks' existing products and provide specialized services (e.g. setting up API gateways) to enable banks and fintechs to access APIs. They also have some requirements for open banking compliance,

- Typically require regulatory permission / license to operate
- Require secure API to receive access to customer data
- Focus on creating standardized and interoperable APIs between banks and fintechs
- Offer technical support with regards to data integration, troubleshooting etc.

5 Monetization framework

Central Bank of Oman proposes monetization framework for all the participants of the open banking. API monetization is important incentive because the participants would adopt open banking in Oman. The monetization framework details out which APIs can be monetized, what are the various monetization models etc.

5.1.1 Classification of APIs for monetization

All the APIs for the use cases should be classified by the participants into three categories, Basic APIs, Standard APIs, Premium APIs.

- **Basic APIs**: These are all the APIs where read-only the information is getting exchanged among the various systems.
- **Standard APIs**: These are the APIs where restricted transactional information like Account details, Customer personal details etc. are getting exchanged.
- **Premium APIs**: These are the APIs where full set of data is getting exchanged. Ex: Credit score, payment initiation APIs, Bill payments API etc.

CBO recommends all the basic APIs not to be charged.

Participants are encouraged to develop their own monetization models but they would be evaluated as part of sandbox testing before launching it in the market.

All the participants should provide monetization strategy to CBO for assessment including but not limited to identification of revenue avenues, API categorization, Monetization model selection and metrics.

Only those monetization model adopted during testing would be allowed to operate in the open market.

5.1.2 Monetization models

1. **Freemium:**

This model allows the consumption of the APIs to a certain threshold value. After which the APIs become chargeable. This is most used model for public APIs. The participants must develop freemium pricing plans, which the consumers can subscribe to.

Example: The first 1000 API calls by the consumer would be free of charge. After 1000 calls, the next calls would be charged OMR 0.005 / call.

2. **API call (Pay per use):**

In this model, the consumers, typically the developers, pay the producer, the ASPSP for each successful call made to the API. Each API call would be charged separately. Another flavor of this would be to charge the consumer per each MB of data used.

This model is best suited when the consumers are already aware of the APIs that are exposed, tried some free APIs and are comfortable with the ecosystem. Also, from the producer end, it would be suited if they have tie up with a digital vendor.

Example: OMR 0.050 per API call. OMR 0.10 per MB of data consumed.

3. **Data Exchange:**

In this model, the bank shares the data each time there is an API call by the consumer. The monetization here could simply be the sharing of data regarding an API request such as details of interest rates for the term deposits or pre-approval details for the customer onboarding. Another variation is the charges are on the data usage.

Ex: OMR 0.05 per MB of data exchanged / shared

4. **Transactions based:**

Transaction-based models are like traditional transactional banking services. The main difference in an API context is the way companies such as PayPal and Stripe allow third parties to integrate and utilize their services through plug-and-play APIs. This allows PayPal and Stripe to reach broader audiences and drive higher transaction volumes to their services. This model assumes that the API is being called to complete some sort of transaction, like paying a bill or transferring money. Users are only charged a fee when the transaction completes.

5. Subscriptions:

Subscription-based models for API access can be either fixed or dynamic. A fixed model is straightforward and offers full API access for a fixed monthly cost. A pay-as-you-go approach is more dynamic, and pricing is determined by metered usage. For example, a cloud computing platform usage price could be determined on an hourly basis by the operating system and platform size. Another dynamic subscription model is a tiered model. Developers' sign-up to and pay for a particular usage tier, based on the number of API calls over a fixed period. While the cost increases per tier, the cost per API call usually drops. For example, Vertical Resources (a process-as-a-service company) uses the tiered business model. Prices per usage drop with consuming higher volumes of API calls and users can adjust their tier based on an analysis of usage over a time. A subscription could be automatically upgraded to the next tier if developers want to continue using the API service after they have reached their subscription limits.

6 Key Performance Indicators (KPIs)

The success of any initiative needs to be quantified. For measuring the success of Open innovation list of key performance indicators that ASPSPs / AISP / PISP must develop and monitor on regular basis. KPIs help in assessing the risks, bottlenecks, and opportunities for the participants to rationalize and optimize the open innovation offerings. These can be bundled into different categories that make up the whole Open banking.

a. Value generated for ASPSPs

i. Total revenue generated by Open Banking APIs monetization.
ii. Revenue per API
iii. Customer reach
iv. Payments initiated by various PISPs

b. Customer satisfaction

i. Availability
ii. Customer satisfaction (through survey)
iii. Active consumers per API
iv. Consent renewals

c. Value generated for AISP/PISP

i. New customer acquisition
ii. Platform availability

iii. Adherence to regulatory requirements
iv. Partner sourced revenue

d. Technical KPIs

i. Response time
ii. API drops
iii. API hit ratio
iv. Authentication miss ratio
v. Outlier transactions

All the KPIs / metrics should be submitted along with other compliance metrics at the frequency prescribed by the CBO.

7 Terminologies

1. **API Management:** Process of creating, publishing, archiving and deleting the web application programming interfaces, enforcing policies, controlling access, collecting and analysing statistics and reporting the performance.
2. **Consent:** Voluntary agreement of the customers to share their personal data with any third party in context of the purpose for which the personal details are shared.
3. **Third party Service Providers (TPPs):** Organizations that interacts with the banks to provide innovative services to the customers.
4. **API Monetization:** Monetizing the APIs by the banks either directly or indirectly.
5. **Participants:** Participants refers to all the entities that are adopting Open Banking ex: Banks, Third party service providers, API platform vendors etc.

8 Glossary

API	Application Programming Interface
ASPSP	Account Servicing Payment Service Provider
AISP	Account Information Service Provider
AML	Anti-Money Laundering
AWS	Amazon Web Services
BAuth	Basic Authentication
BFSI	Banking, Financial Services and Insurance
CBO	Central Bank of Oman
CEG	Customer Experience Guidelines
CI/CD	Continuous Integration and Continuous Deployment
CSS	Cascading Style Sheets
DDoS	Distributed Denial of Service
DevOps	Development Operations
DevSecOps	Development, Security and Operations
eIDS	Electronic Idenitities
ERP	Enterprise Resource Planning
FAPI	Financial Grade Application Programming Interfaces
FI	Financial Institutions
HMAC	Hash-based Message Authentication Code
HTTP	Hypertext Transfer Protocol
HSM	Hardware Security Module
IAM	Identity and Access Management
JSON	JavaScript Object Notation
JWT	JSON Web Tokens
KYC	Know Your Customer
KPI	Key Performance Indicator
MFA	Multi-Factor Authentication
OAS	Open API Specifications
OIDC	OpenID Connect

OMR	Omani Rials
PISP	Payment Initiation Service Provider
PSD2	Payments Service Directive 2
PSU	Public Sector Units
REST	Representational State Transfer
SAML	Security Assertion Markup Language
SLA	Service Level Agreements
SOAP	Simple Object Access Protocol
SSL	Security Sockets Layer
SSO	Single Sign On
TLS	Transport Layer Security
TPP	Third Party Provider
UI /UX	User Interface / User Experience
URL	Uniform Resource Locator
WAF	Web Application Firewall
YAML	Yet Another Markup Language
XML	Extendible Markup Language