FINANTEQ®

Google Pay and Apple Pay Provisioning and Verification Software Development Kit (SDK)

Component Overview

CONFIDENTIAL

FINANTEQ hereby restricts the confidentiality of all information contained in this document and obliges you not to reveal any content to third parties in whatever form.

List of contents

1. Introduction

The growing popularity of Google Pay and Apple Pay strongly encourages banks all over the world to implement these solutions. In order to do so, a bank needs to comply with different requirements imposed by Google and Apple. Two important requirements to meet are: the ability to initiate the Card Provisioning process from the bank's iOS or Android application and the ability to Verify the user and activate a payment pass using the bank's iOS or Android application.

However, the in-house implementation of these requirements is not so straightforward.
Here are the three main reasons why:

1. In order to create and implement the above-mentioned functionalities the bank has to continuously **collect necessary business requirements from several parties involved in the process**, i.e. Google, Apple, MasterCard (MDES) and Visa (VTS). Factors such as imprecise documentation or the analysis and **decisions regarding possible implementation variants** (e.g. in the case of In-App Verification) are also very important here. We have gone through this process and we know empirically that the technical part of the implementation is only one part of the puzzle. The key to implementing the solution in the optimum time and without incurring additional costs is the know-how. Lack of knowledge prolongs the implementation process even by several months. Examples of what may happen, what decisions the bank will have to take and how to prepare for the test can be found in the Benefits section of the know-how paragraph.

2. If a bank wanted to create both components (Apple Pay and Google Pay) on their own, then we estimate that the analysis, design, development, testing and deployment therefore would take about **200 MD** if done by a team without the know-how in this matter (estimation does not include development of user interface).

3. Before the bank can deploy a solution on the production environment, it has to pass tests conducted by external auditing companies and be approved by Apple and Google, which further complicates the process. A bank that does not pass the tests will have to implement the improvements imposed by Google or Apple. Every time the bank will also have to cover the high cost of taking the test – usually **several thousand Euro per attempt** – and wait another few weeks for test results.

Summing it up: There is a risk that the bank will spend months building the solution and collecting business requirements and know-how from parties involved, and the solutions may not be accepted in the end. Additionally, each failure to meet the test requirements forces the bank to repeat the expensive and long audits.

One advice for free: **Do not promise anyone any specific dates or deadlines for implementing Google/Apple Pay, especially in public.**

This is why (click to open link):
5-challenges-a-bank-will-encounter-while-implementing-apple-pay-and-google-pay/

2. Component description

To make the whole situation as seamless as possible for banks, we created **Google Pay and Apple Pay Provisioning and Verification SDK** (Software Development Kit). The solution enables the bank to implement in their mobile application the functionality of initiating the Card Provisioning process through the bank's application and the ability to verify the user and activate a payment pass using the bank's application. It should be noted that these components are 100% tested and compliant with Apple and Google requirements and, owing to the use of native SDK libraries, they are also easy to integrate with the bank's application.

In order to further simplify the implementation of Apple Pay and Google Pay, our component has also been equipped with a library for generating payload. The payload generation process requires specialised knowledge of cryptography and proper application of cryptographic standards. If the bank's development team does not have competence in the areas of PKI, symmetric and asymmetric cryptography and NIST cryptographic standards, we recommend using a ready solution, since inadequate payload preparation may result in time-consuming search for the cause of errors. In addition, our library supports all the data formats required by Mastercard and Visa necessary to generate the payload.

Our components will make it easier for the bank to fulfil the agreement with Google or Apple obliging the bank to implement Provisioning and Verification into the mobile banking application in the given time. Thanks to our components, the bank will not have to burden its development teams and detach them from other tasks.

Our components have been created for development teams that want to implement Provisioning and Verification processes in banks quickly and without unnecessary risks.

3. Functionalities and Features

Apple Pay SDK

- checking the availability of Apple Pay,
- getting cards to activate,
- activating a card (In-App Verification),
- adding a card to Apple Pay (In-App Provisioning),
- configuring the “add to Apple Pay” button visibility and handling the tap action,
- getting device names for activated payment passes.

Google Pay SDK

- checking the availability of Google Pay,
- checking the status of the wallet,
- checking the status of cards in the wallet:
  - checking a single card,
  - checking the list of cards,
  - checking the list of cards and store their statuses,
- adding a card to the wallet (Push Provisioning),
- activating a card (App-to-app Verification),
- removing a card from the wallet,
- setting Google Pay as the default contactless payment application.

Server Wallet SDK

- Interfaces and data model for:
  - Apple Pay payload service,
  - Google Pay payload service,
  - Wallet OTP service.

Library for generating the card payload

- Generating the card payload in the selected data format,
- Generating the value used to authorize card provisioning and verify the user adding the card from the wallet level.

4. Benefits

Time saved on collecting know-how – we give the bank a knowledge about the correct implementation of processes in their applications to make the deployment as quick as possible. We deliver a ready to deploy SDK, but also provide the bank with invaluable Know-How Documentation which accelerates the implementation by a **several months**. Below we listed a few examples of such knowledge:

a. We will advise which elements can be reused without paying twice for them.
b. We will advise in terms of which of the available In-App Verification paths should be adopted based on the bank's needs.
c. We will advise which card identifications sent by the server are needed to correctly perform the Provisioning function.
d. We will help to avoid unnecessary processing and errors associated with data conversion in the Provisioning and Verification processes.
e. We will advise the bank how to test Provisioning and Verification before publishing the application on production (as the documentation you receive from one of the parties may be confusing).
f. We will advise how to correctly configure the project so that Apple Pay and Google Pay work without any problems.
g. We will advise about what are the formalities related to the publication of the application in the stores to save time on the time-consuming bureaucracy.
h. You will get information about the functions that you can skip and those which are necessary to be implemented to pass the tests/audits.
i. You will save time determining the format of the data used to identify Google Pay cards during the App-to-app Verification process.

Time saved on development — our component shortens the implementation time of Provisioning and Verification in the mobile banking applications usually twice, compared to a situation where the bank builds the solution in-house.

The highest standards — our solution is 100% compliant with Apple and Google requirements concerning Provisioning and Verification. We provide the knowledge about effective implementation of these processes in applications, which makes it possible to pass the tests carried out by Apple and Google before such solutions are made available on the production environment.

Native SDK libraries — easy to integrate with APIs of bank applications and not restricting the user interface design, which enables the bank to design the UI in accordance with its own concept.

Two components, one bundle — you do not have to look for two separate suppliers for Google Pay and Apple Pay. We offer a bundle including both these components which can be implemented in any order, according to your needs.

Source code — we can provide the bank with the source code of the components.

Automated testing – we provide the solution with automated testing included.

Minimum work on the part of the bank — the components include a pre-defined integration with the system wallets of Google and Apple on the mobile side.

5. Architecture

Each component was created in the native language of a given platform, thanks to which it can be easily implemented in each of bank's mobile banking systems.

Platform	Programming language
Google Android	Kotlin, Android API 19.0 and above
Apple iOS	Swift 5.0, iOS 10.0, Xcode 10.2 and above
The server application	Java 1.8/Kotlin

6. Requirements

In order to fully implement the Provisioning (In-app/Push) and Verification (In-App/App-to-App) processes of Apple Pay and Google Pay, the bank should:

1. Sign the necessary agreements with Apple, Google and the suppliers of tokenization services (e.g. Visa, Mastercard).
2. Provide services generating encrypted payment data payload and cryptographic OTP/TAV for activated cards in accordance with the requirements of particular PNOs (Payment Network Operators).
3. Implement the necessary changes in the application UI (presenting the *Add to Apple Wallet* / *Add to G Pay* button in accordance with Apple/Google guidelines) and notifications enabling the transfer of the necessary information in the Provisioning and Verification process between the mobile application and the server in the Bank.

7. Distribution and Business Model

The components are available as a set of libraries and SDKs along with the whole technical documentation detailing all aspects of the integration and source code.

Business model:

1. One-off perpetual license fee per bank per country.
2. Implementation support.
3. Optional SLA/maintenance if required by bank.

8. Build Your Mobile Banking From Components

Fast time to market is the key to win today. This is why we present to you FINANTEQ's components – a quick and tested way to accelerate your time-to-market.

- All of our components are SDK's and can be easily imbedded into any existing app (made by FINANTEQ or any other IT provider)
- We provide detailed technical documentation for developers.
- Components are ready to blend into any mobile banking design.

Discover the wonderful world of our components and make your mobile banking modular!
Visit: https://finanteq.com/components/

List of FINANTEQ’s products and components

Product	Description
Finanteq mBanking	A component solution for mobile banking that allows users to manage their finances from the mobile application level. It consists of the mobile part - application for Android and iOS, the server part (Java application), which integrates with services on the bank' s side, the Backoffice application enabling mobile channel management, as well as infrastructure ensuring effective provision and implementation, taking into consideration e.g. test automation, simulation of the banking environment (or services not yet available), configuring functionalities, etc. The functionalities are customized for retail, SME and corporate banks.
Supewallet (mCommerce/VAS)	An m-commerce platform which enables any mobile banking application to be transformed into a superwallet. The platform offers the possibility of buying directly through the banking application, e.g.: bus tickets in over 100 cities in Poland, paying parking fees in several dozen cities, buying gifts, sending flowers, booking hotels, ordering and paying for taxi services.
Extentum (new)	A no-code platform for quick extension of functionalities in already existing banking applications (including internet banking) by non-technical business persons. All IT/DEV processes have been eliminated. A typical ~300 man-day functionality with delivery time of 4-6 months, normally created by a multi-person team of programmers, can be built with Extentum by one business person within 1-2 days and implemented without changing the version.
Pocket Branch	Pocket Branch enables creation of virtual bank branches. It is an omnichannel solution that facilitates communication between the bank and the customer through: chat, audio or

	video conversation. With Pocket Branch, just like in a physical branch, customers can obtain information, help and even conclude and finalize transactions. All of this is available through the customer's mobile application or the bank's website. The solution is offered as a whitelabel or SDK (software development kit), which makes it compatible with any mobile banking applications.
Finanteq Mobile Stock Exchange	A mobile platform for trading in securities markets. It enables the user to view current prices, ratings and market information.
Finanteq Mobile Currency Trading	An application that enables performing transactions on the currency trading market in real time. The module allows the user to place orders for currency exchange at a specific rate, as well as to notify the users about achieving the assumed rate by the selected currency pair. The application also enables access to historical data in the form of charts and to the history of all conducted currency transactions.
Finanteq mToken	An application installed in the phone memory which is used to generate one-time passwords in order to secure on-line and mobile transactions.
Smartwatch Starter Kit	A smartwatch application that consists of an SDK that enables implementation of the solution in any mobile banking application of a third party. The functionality includes: displaying the account balance, geolocation of points (ATMs, branches), displaying currency pairs and reading PUSH messages.
OCR Module	The module allows scanning any data with the phone camera, including entire documents such as invoices or ID cards. The advantage of the solution lies in the off-line processing of sensitive data without sending it to the cloud.

Google & Apple Pay Module	Available as an SDK along with documentation. It gathers all the necessary code and know-how required to add a payment card to Apple and/or Google wallet in a banking application.
Push Notification Server	A system for managing and initiating Push notifications.
Productive24 Platform	A low-code platform for building applications that support any internal and external processes of banks. An eLeader Group product; Finanteq is part of eLeader Group.

9. About FINANTEQ

We help banks and fintechs win in mobile.

FINANTEQ is a software and consulting company focused on mobile finance. We provide complete mobile solutions as well as ready-to-integrate add-on platforms and components for the mobile channel.

According to Deloitte, FINANTEQ is ranked among top fintech companies in Poland. Our innovations, for example the SuperWallet, have received awards in prestigious global competitions such as “Finovate New York” and “Citi Mobile Challenge EMEA”.

Clients: BNP BGŻ Paribas, Santander, Danske Bank, Unicredit, National Bank of Kuwait, mBank. Also, FINANTEQ’s mobile banking application for Santander Bank was ranked #3 in the world by Forrester Research.

10. Contact

Contact us and find out how our component can enhance your mobile banking.

e-mail:
contact@finanteq.com

phone:
+48 81 536 9692