

Cruise Mobile Apps

2026 Annual Report

OnDeck by sourcetoad

Contents

Introduction 3

Scope 4

Methodology 4

Mobile Applications 6

Features 8

Update Frequency 13

Tools & Key Players 14

Security 29

Rating our Predictions from 2024 32

Looking Ahead 34

Conclusion 38

About OnDeck

OnDeck by Sourcetoad has specialized in innovative guest-facing software for the cruise and ferry industry since 2010. Our mission is to increase the value of both the guest's and the crew's time by building automation, communication, and self service systems.

We offer a number of custom and out-of-the-box solutions for cruise lines to increase operational efficiencies, improve guest experiences, free up crew schedules, and shorten technology ROI cycles. We offer the following products and services to our customers:

Cruise Director Platform

- Shipboard and shoreside middleware and API management
- Ship-to-shore data synchronization
- Shipboard CDN system
- Onboard event analytics
- Infinite level permissioning and user management

Mobile App-in-a-Box

Our guest mobile app works with the Cruise Director Platform to provide guests with everything from digital check-in to dinner reservations and folio management. Compatible with Otalio, MXP, and Seaware.

Platform Modules

- **Entertainment**: music, TV, VOD, streaming, management, and iTV systems
- **Intelligent Signage**: Advanced onboard digital signage and interactive kiosks
- **Captive Portal**: iCafe systems, and guest/crew wifi access management
- **Digital survey**: Automated digital surveys for pre-cruise, post-cruise, spa, dining, and ShoreEx reviews for app and iTV
- **Push notifications**: A full push notification module for mobile apps and iTV
- **ECS**: Our workflow-based, onboard passenger notification system

Professional Services

OnDeck can help you develop the next groundbreaking solution in the Cruise industry.
Our professional services include:

- Shipboard and shoreside AI integration and development
- Custom software engineering for land and marine operations
- Documentation as a service
- 24/7/365 help and support desks

Introduction

We have been producing this report since 2019, and each edition has had its own character. The 2022 report was about recovery and catching up after two years of pandemic-era stagnation. The 2024 report was about AI arriving at the door and an industry that wasn't quite sure whether to let it in. This year's report is about something different: execution.

The apps in this cohort are, on the whole, more mature than anything we have analyzed before. Frameworks that were experimental two years ago are now production mainstays. Migrations that had been in progress for years have finished. The fleet has grown to 21 apps, including several new entries from lines that have finally decided a passenger-facing mobile app is no longer optional. And under the hood, some of the most interesting architecture decisions we have ever seen in this space are starting to take shape, quietly, in the dependencies and manifests of apps that look from the outside like incremental updates.

That said, this is not a story of universal progress. The security findings in this report are, in some respects, more concerning than those in 2024, not because the problems are new but because they are the same problems, still present, still unaddressed in parts of the fleet. Accessibility remains an afterthought for most apps. And the gap between the leaders and the laggards, in features, in technical quality, and in security hygiene, is wider than it has ever been.

What is genuinely new this year is AI. Not as a prediction or a prediction of a prediction, but as something that is actually shipping. One major brand family has made a coordinated platform commitment to AI-assisted guest services. A virtual concierge deployed across part of one fleet handled over 13,000 service requests in six months. Additionally, the infrastructure for in-app voice calling is being wired in across multiple operators simultaneously. The AI chapter of cruise app development has started, and it started faster than most of the industry expected.

This year we also used AI in our own process for the first time, to help analyze the apps themselves. We have written about that in the Methodology section, and we think it is worth reading if you are curious about how this kind of analysis is going to work in a world where the volume of material grows faster than any human team can manually process.

As always, this report reflects our honest read of what we found. We call out good work when we see it and problems when we find them. We hope it is useful, and we hope the right people are reading the Security section in particular. Enjoy.

New A new addition to the 2026 report.

Updated A significantly updated section.

Scope *Updated*

In this year's report, we focused exclusively on mobile apps designed for ocean cruise passengers. The apps included in our analysis met the following criteria:

- They are the primary guest-facing mobile app used by passengers before and during their cruise experience.
- They are used across multiple ships within an ocean cruise fleet.
- They are downloadable apps available on both Android and iOS.
- They are not web-based (Progressive Web Apps), crew-only apps, brochure apps, or supplemental entertainment applications.
- They were released and tested before January 1st, 2025.

We excluded cruise lines whose primary focus is river cruising, as well as those whose only digital experience is delivered through web-based apps.

A Note on OnDeck's Role in This Report

OnDeck by Sourcetoad, the publisher of this report, develops and maintains the Cruise Director Platform and the OnDeck Mobile App, which together serve as the underlying middleware, app infrastructure, and guest-facing mobile experience for several cruise lines analyzed in this cohort. In this edition, three apps in the report are built on these platforms. We have applied the same analytical methodology to these apps as to all others in the cohort, and no findings related to these apps have been omitted or softened as a result of that relationship.

Methodology *Updated*

A Note on Progressive Web Apps and Onboard-Only Web Applications

Some cruise lines have chosen to deliver their guest-facing digital experience through Progressive Web Apps or onboard-only web applications rather than downloadable native apps. These solutions can replicate many of the same features as a native app, but their onboard-only accessibility places them outside the scope of this analysis, which focuses on apps available to guests both before and during their voyage.

Our Process

Our methodology has been consistent since we started this report in 2019, and has evolved each year as the apps themselves have become more complex to analyze. The core process remains the same: we obtain the Android APK for each app in the report, decompile it, and examine what is inside.

For apps that are available in the US region, we download them directly onto a Samsung Android device. For apps that are not available in the US, we sideload them via ADB. Because Android apps are increasingly distributed as split APKs rather than a single unified package, we pull the English-language ARM64 base and feature splits and merge them into a single unified APK using APKEditor before analysis begins. This ensures that every app in the report is analyzed on a consistent basis regardless of how it was originally packaged.

From there we decompile the unified APK using Apktool, which gives us a human-readable view of the app's manifest, resources, assets, and configuration files. We then run each app through MobSF, a mobile security analysis framework that checks for common vulnerabilities and flags security concerns automatically. For apps that appeared in a previous edition of the report, we run a diff against the prior year's decompiled output using Araxis Merge, which lets us see exactly what has changed between versions. For apps that are new to the report, we examine the decompiled output directly in VS Code.

What's New in 2026

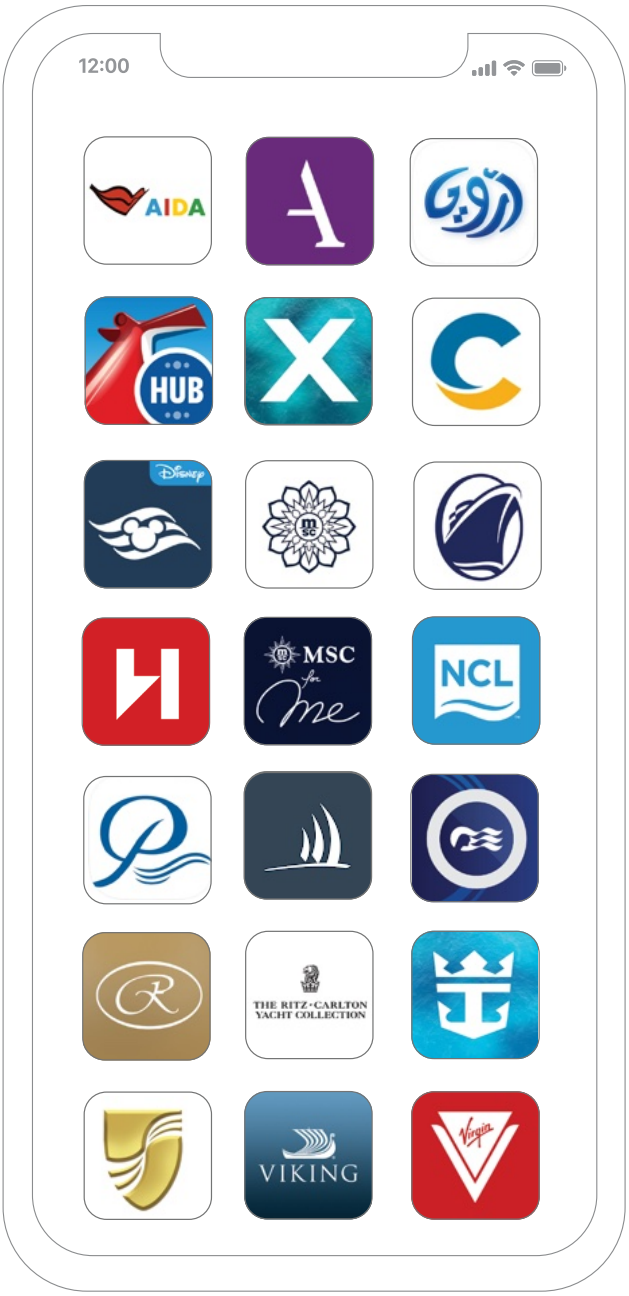
Two things changed meaningfully in how we worked this year. First, the number of apps grew. With 21 apps in the cohort and several new entries, the volume of material to analyze increased significantly. Second, and more importantly, we incorporated AI agents into the analysis workflow for the first time.

In previous years, the comparison writeups that underpin the Tools and Key Players, Security, and other analytical sections were produced entirely by human analysts working through the decompiled output manually. This year, we built a structured analysis repository around the APK workspace that allowed AI agents to work directly from the decoded artifacts. Each agent was given a consistent set of rules for interpreting file changes, a template for structuring comparison documents, and access to the decoded folders for both the current and prior year's APK. The agents produced first-draft comparison writeups that human analysts then reviewed, corrected, and expanded.

The output of that process is what you are reading. The analytical judgments in this report, including what we call out as significant, what we treat as noise, and what we flag as a security concern, reflect human review of AI-assisted analysis rather than either purely human or purely automated output. We think this is the right way to describe it, and we think it is the direction the industry as a whole is heading.

If you would like your app or a feature included in the next report, please contact us at **appreport@sourcetoad.com**.

Mobile Applications



AIDA Cruises

AIDA Cruises was released in May 2012 and works across the full AIDA fleet.

Ambassador Cruise Line

Ambassador Cruise Line was recently released in September 2023. The app works on the full Ambassador fleet, with feature availability varying by ship.

AROYA Cruises New

AROYA Cruises was released November 2024, and works onboard the *AROYA* ship.

Carnival Cruise Line

Carnival HUB was released in January 2015 and works across the cruise line's entire fleet.

Celebrity Cruises

Celebrity Cruises was released in November 2017. The app now works on on the full Celebrity fleet.

Costa Cruises

Costa App works across the full Costa Cruises fleet. It was released in January 2018.

Disney Cruise Line

Disney Cruise Line Navigator was originally released in June 2013, and had a major redesign in February 2018. The app works on the entire Disney fleet.

Explora Journeys

My Explora was released in June 2023. The app works onboard the *Explora I* and *Explora II*.

Holland America

Holland America Line Navigator was released in February 2019 and works across the cruise line's entire fleet. The app shares a code base with Seabourn Cruise Line's app, *Seabourn Source*.

Hurtigruten

Hurtigruten's onboard app was released in June

2019. The app is supported on ten ships in the Hurtigruten fleet (*MS Otto Sverdrup, MS Nordlys, MS Nordnorge, MS Richard With, MS Nordkapp, MS Polarlys, MS Kong Harald, MS Vesteralen, MS Trollfjord* and *MS Midnatsol*).

MSC Cruises

MSC for Me was released in May 2017 and works across the entire MSC Cruises fleet, with features and functionality varying by ship.

Norwegian Cruise Line

Norwegian Cruise Line's updated app was released in August 2024, and works onboard the full NCL fleet.

Paul Gaugin New

The Paul Gauguin Cruises Travel App was released in August 2024, and is available on the entire fleet. It shares a codebase with the Ponant app.

Ponant

Released in January 2019, *Ponant* works across the cruise line's full fleet.

Princess Cruises

Princess Cruises was released in November 2023 to replace the *MedallionClass* app. It works on the entire Princess Cruises fleet.

Regent Seven Seas Cruises

Regent Seven Seas Cruises was released in September 2022. The app works onboard the *Seven Seas Grandeur*. This app is a departure from sharing a code base with other NCLH brands.

Ritz-Carlton Yacht Collection

Yacht Journey was released in July 2022. The app works onboard the *Evrima*.

Royal Caribbean International

First released in November 2017, *Royal Caribbean International* is available across the cruise line's full fleet. The app shares the same code base as the *Celebrity Cruises* app.

Seabourn Source

Seabourn Source went live in April 2020 and works across the cruise line's whole fleet. The app shares a code base with *Holland America Line Navigator*.

Viking Ocean Cruises

Viking Voyager was released in June 2017, and the app is available on all active Viking Ocean, River, and Expedition ships.

Virgin Voyages

The *Sailor* app works on all four ships in the Virgin Voyages fleet. The app went live in February 2020.

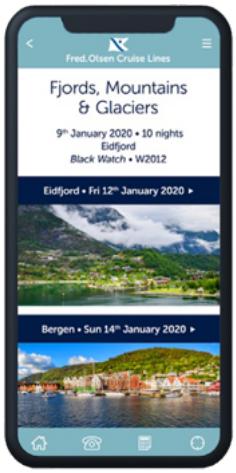
Features

Updated

To help us see the big picture trends, we've categorized features into three groups: **Common** (a feature present in nearly every app), **Less Common** (present in a minimum of three but not in more than half of the applications), and **Stand-Out** (unique to only one or two applications).

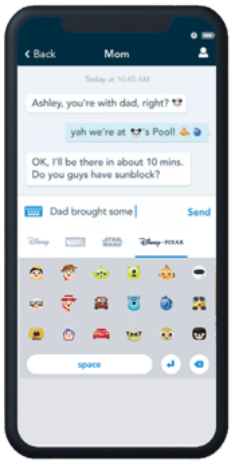
Common Features

The following features are shared by a majority of the apps we analyzed.



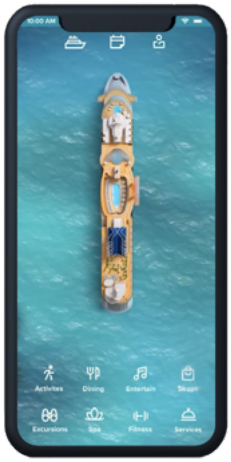
Activities & Excursions

Activity and excursion information is a universal feature across onboard apps, with most apps allowing guests to view and book excursions while onboard. Some offer the option to pre-register or book for activities before the cruise.



Chat

Chat has become a common feature of onboard apps. Some apps offer a group chat option in addition to one-to-one chat.



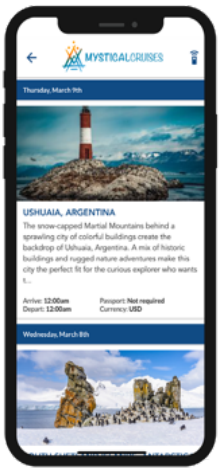
Deck Map

Ship maps and directories are another ubiquitous feature of onboard apps. The more robust deck maps are interactive and searchable, making it even easier for passengers to navigate the ships.



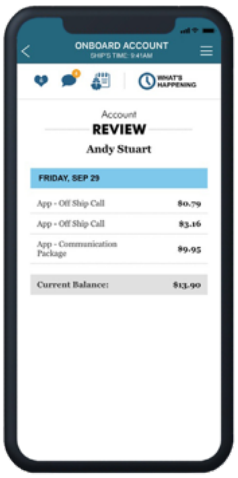
Dining

A majority of apps have information about dining options onboard the ship, including menus, hours, location, and dress code. Many apps also offer reservations, with some opening bookings prior to departure. One app offers a virtual queuing option.



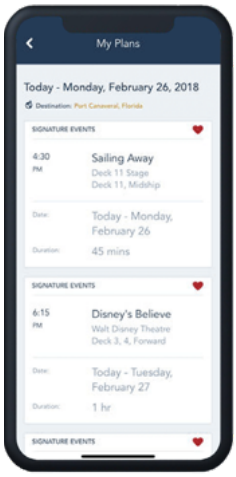
Destination Information

Most apps have now added port and destination information—including photos, activity suggestions, and cultural and historic information—allowing guests to learn about the ports of call along their cruise route.



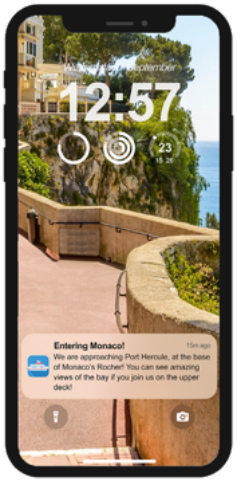
Folio

Nearly all onboard apps have a folio feature, which serves as a detailed billing record that refreshes as new charges are applied to a user's account. This feature enables passengers to see all the expenses billed to their account in real time.



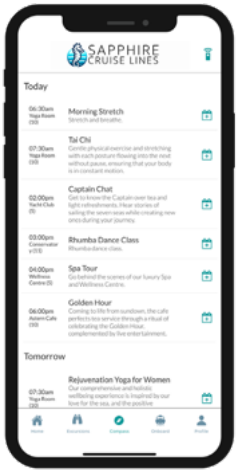
Itinerary

The itinerary is the passenger's personalized schedule. Activities that have been marked as favorites or confirmed with a booking appear on the guest's itinerary. One app offers the option to link itineraries with other guests so they can easily coordinate plans.



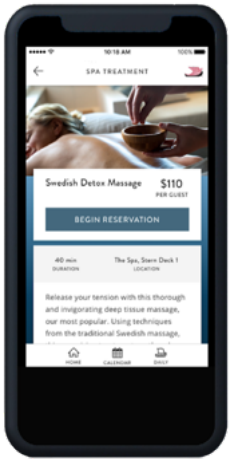
Notifications

Push notifications have become a standard feature in passenger-facing apps. Guests can receive notifications about their scheduled activities, updates to their billing folio, new chat messages, and more. As with onboard chat, user reviews reveal mixed success of this feature.



Schedule

The daily schedule is the hub for organizing daily activities. In most apps, events can be added, reserved, or highlighted before being copied to the guest's personal itinerary.

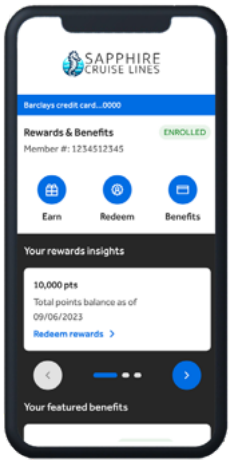


Spa & Wellness Services

Most onboard apps offer information about the spa services available during the cruise, and many allow guests to book appointments. Some apps have expanded this feature to work shoreside, allowing guests to make reservations prior to departure.

Less Common Features

The following features are found in at least three, but no more than half of the apps we analyzed.



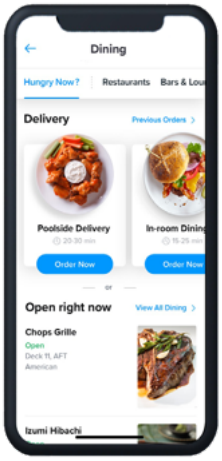
Credit Card Approvals

A few apps have integrated the option to apply for instant approval of a cruise-line branded credit card. The card is then added to the guest's digital wallet, so they can begin spending and acquiring loyalty points immediately.



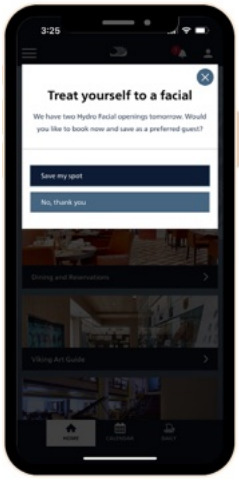
Digital Photo Archive

Some apps offer guests the option to review, purchase, and share photos from the cruise.



Mobile Ordering

Several apps have mobile ordering features, offering guests the option to order food, beverages, or other services delivered to the guest's stateroom. Two apps offer location-based delivery anywhere onboard.



Promotions & Personalized Offers

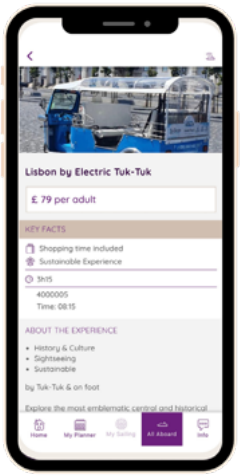
New

Targeted deals, discounts, or package offers delivered through the app based on user behavior or profile data.



Ship Tracking

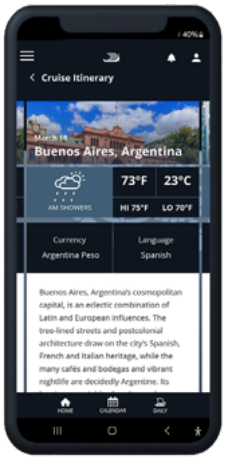
Live ship tracking allows guests to see the ship's location in real time. One app also offers a live onboard video feed.



Shoreside Bookings

New

This year, more apps have added the option to book or pre-register for shore excursions, spa treatments, and restaurant reservations in the pre-cruise environment.

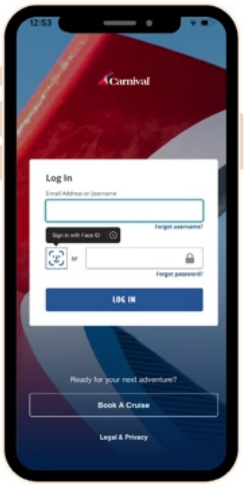


Weather

Weather features allow guests to view the forecast for each day of their cruise, helping them plan appropriately for shoreside excursions and expeditions.

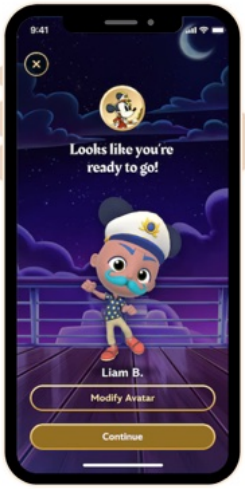
Stand-Out Features

Below are some of the unique features that stood out to us.



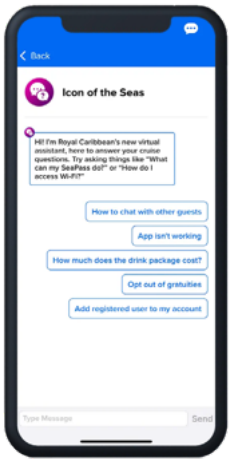
Biometric Login *New*

A few apps have added support for biometric authentication, such as Face ID or fingerprint recognition, allowing users to securely log in and access their account without entering credentials.



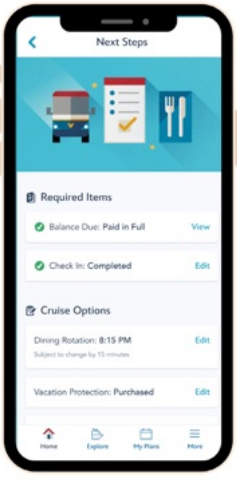
Gamified Experience *New*

One app includes an augmented reality experience that uses the phone's camera and onboard sensors to create an interactive, ship-wide activity.



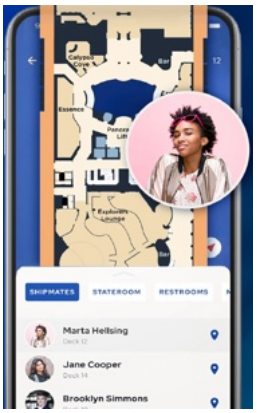
Guest Services Chat

A few apps offer the option to chat directly with guest services. Two apps offer a chatbot virtual assistant that can answer basic questions, provide information about the ship, or connect the guest with a live agent.



Integrated Travel Companion *New*

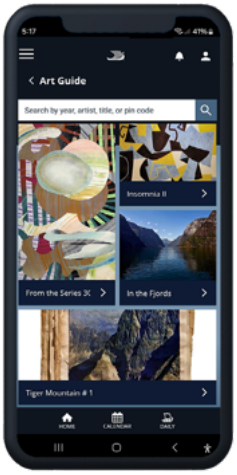
A small number of apps extend beyond the cruise itself to support the full travel journey, including flights, accommodations, and transportation.



Live Passenger Location Tracking

New

One app offers users the option to view the real-time location of other guests in their travel party, along with wayfinding. This enables easier coordination onboard by helping passengers find and connect with one another across the ship.



Virtual Art Guide

Two apps now include virtual art guides. These work in fairly different ways, but are interesting and unusual features none the less. Guests can listen along to audio clips to learn historical information about the works of art onboard.

Update Frequency

Updated

The update frequency data reflects an industry that is actively maintaining its applications, but not uniformly advancing them. Several leading apps are pushing updates at a sustained cadence, often exceeding a dozen releases in 2025, tied to meaningful feature expansion and platform evolution. At the same time, a many apps update infrequently, with some seeing only a handful of releases and at least one showing no updates at all. The gap between actively developed apps and those in maintenance mode continues to widen.

Update frequency alone is no longer a reliable indicator of maturity. High-velocity apps are often in the middle of significant rebuilds or architectural shifts, while lower-frequency apps tend to be stable but largely static. The defining difference is intent. The apps that are improving show sustained, purposeful iteration, while others exhibit clear signs of stagnation, reinforcing the growing divide between apps leading the pack in innovation and those that are lagging behind.



Tools and Key Players Updated

Two years is a long time in mobile development, and our analysis of this year's group of apps reflects it. The most interesting story in this year's tools analysis is not any single SDK or framework, it is the pattern of consolidation. Brand families are making coordinated platform decisions, and teams are completing migrations that have been in progress for years. What follows is our attempt to document that transition as it is actually happening.

New to 2026 New

The 2026 analysis surfaces a notably wide range of first-time arrivals across the apps. Some represent coordinated platform bets by brand families, others appear independently in apps that share no obvious lineage. A few are surprising. Taken together, they point to development teams that are simultaneously hardening security posture, expanding communication stacks, and getting more serious about identity verification at check-in.

Adobe Experience Platform Mobile SDK

Adobe — *developer.adobe.com*

Adobe's Experience Platform Mobile SDK is appearing in several apps at once, which is rarely a coincidence. The SDK acts as a unified collection layer for analytics, audience segmentation, and campaign targeting. In the cruise context, this is almost certainly being used to replace or augment legacy analytics tools and pull together a more cohesive picture of guest behavior across shoreside and onboard touchpoints. Its simultaneous appearance across multiple brands suggests either a coordinated push from a shared technology vendor or that a number of teams have independently arrived at the same conclusion about first-party data strategy.

Akamai Bot Manager

Akamai — *akamai.com*

Akamai Bot Manager is a security layer that detects and blocks automated or malicious traffic. Its native library appearing across several cruise apps points to a growing concern about scrapers, credential stuffing, and automated booking abuse. It is notable that the SDK is embedded at the native level rather than deployed purely as a server-side filter, suggesting that some of the bot detection logic is running on the device itself. This is an unusual and relatively sophisticated approach to mobile security.

Android Privacy Sandbox / AdServices

Google / Android — *developer.android.com*

Google's Privacy Sandbox permissions are appearing in nearly every app we analyzed in 2026. This is largely a platform-level change rather than a deliberate vendor choice: the ACCESS_ADSERVICES_ATTRIBUTION and ACCESS_ADSERVICES_AD_ID permissions are being pulled in by updated versions of analytics and advertising SDKs that have added Privacy Sandbox support to comply with new Android requirements. The presence of these permissions does not necessarily mean every cruise line is running mobile advertising campaigns. In many cases it simply means they updated their Firebase, Braze, or analytics library dependencies and this came along for the ride.

Apptentive

Alchemer — apptentive.com

Apptentive, now owned by Alchemer, is an in-app feedback and ratings tool that allows developers to trigger review prompts and satisfaction surveys at contextually relevant moments. Its appearance in multiple apps suggests that cruise lines are getting more deliberate about collecting guest feedback during the experience rather than waiting for a post-cruise email survey. Timing these prompts well in a cruise context could be tricky, though. There is a meaningful difference between catching a guest immediately after a smooth embarkation and catching them midway through a three-hour tender queue.

CardinalCommerce (3DS2)

Visa/CardinalCommerce — cardinalcommerce.com

CardinalCommerce is Visa's 3D Secure 2.0 authentication solution, designed to reduce fraud during card-not-present transactions. Its appearance in a cruise app likely reflects the addition of in-app payment flows that require bank-level authentication, particularly for higher-value pre-cruise purchases or onboard credit additions. 3DS2 is generally far less disruptive to the user than the original 3DS protocol and is invisible in many cases. Its presence in the binary confirms that at least one app is now handling authenticated card transactions directly within the app rather than deferring entirely to tokenization.

Didomi

Didomi — didomi.io

Didomi is a Consent Management Platform used to collect and record user consent for data processing under GDPR and similar regulations. It is the primary consent tool in a pair of apps sharing a common codebase under a shared package namespace, and was already present in the Rivers version of this report. Its arrival in the Cruise fleet audit reflects the growing number of European-market brands in the report. One brand's 2026 build significantly expanded Didomi's string resources by nearly 190 keys, suggesting a more thorough and legally complete consent implementation than was present in 2025. Didomi represents an interesting contrast to OneTrust, which appears elsewhere in the fleet for similar purposes.

Dynatrace

Dynatrace — dynatrace.com

Dynatrace is an application performance monitoring platform that can be deployed into an Android app by simply replacing the root application class. Its appearance as the application wrapper across three apps from different brands is notable, and suggests these lines may share a common technology management layer, possibly delivered through a shared platform vendor. Dynatrace provides AI-assisted root cause analysis, real user monitoring, and infrastructure observability. For a development team trying to diagnose issues on a ship in the middle of the Pacific, the ability to get automated root cause analysis from a remote monitoring tool is genuinely useful.

Flutter Local Notifications

MaikuB / Open Source — pub.dev

Flutter Local Notifications is an open-source library for triggering local push notifications from Flutter

apps. Its inclusion in one app is a relatively low-key addition on its own, but it fits into a broader pattern of onboard notification infrastructure expanding across the fleet. Unlike remote push systems that require an internet connection, local notifications can fire entirely offline, which makes them well-suited for time-sensitive onboard messaging like activity reminders or dining reservation alerts.

IndoorAtlas

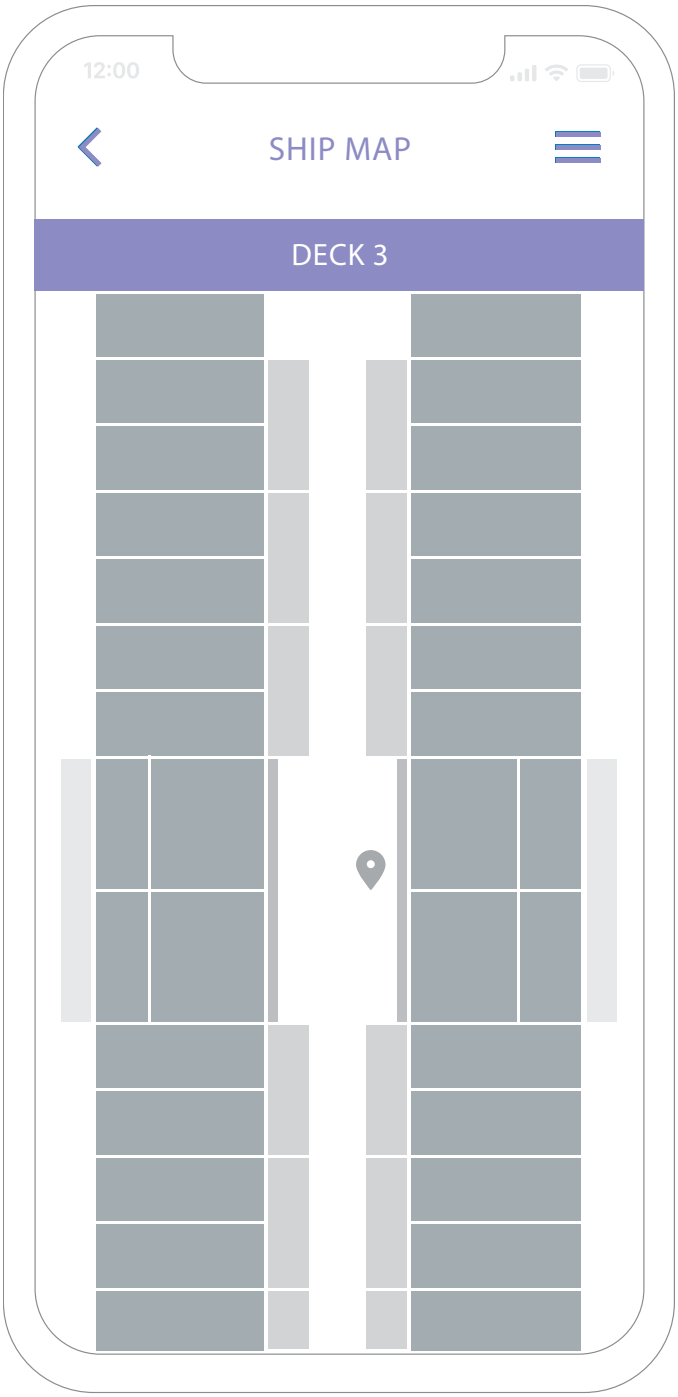
IndoorAtlas — *indooratlas.com*

IndoorAtlas is a Finnish company that provides indoor positioning using magnetic field mapping rather than GPS or Bluetooth beacons. Their SDK appearing in one large fleet operator's app is one of the more ambitious finds in this year's report. Indoor positioning on a cruise ship is a significantly harder problem than in a shopping mall or airport, given the steel structure of the vessel and the constantly shifting magnetic environment at sea. If IndoorAtlas is being used at full scale on these ships, it would represent one of the most complex indoor navigation deployments in the industry. We suspect it may still be in an experimental or limited rollout phase, but it is worth watching closely.

Isar

Simon Leier / Open Source — *isar.dev*

Isar is a fast, embedded NoSQL database for Flutter applications. Its addition to one luxury line's app suggests a move toward more offline-first data storage, possibly to improve app responsiveness when onboard connectivity is limited. Isar is considerably faster than SQLite at scale, which could matter for apps that need to store and query large sets of voyage data, itinerary content, or activity schedules locally on the device.



Jetpack Compose

Google / Android — *developer.android.com*

Jetpack Compose is Google's modern declarative UI framework for native Android development. Its appearance in two major brands' 2026 builds — one

a large contemporary line undergoing a significant Compose refactor, the other completing a full React Native to native rebuild — reflects a broader native Android modernization trend happening somewhat quietly alongside the more dramatic cross-platform migrations. Compose often enters a codebase through rewrites and library updates rather than as a deliberate new vendor decision. Its presence in these builds suggests that native Android development is not disappearing from cruise apps. It is simply being modernized.

Keycloak

Red Hat / Open Source — *keycloak.org*

Keycloak is a well-established open-source identity and access management system developed by Red Hat. Its appearance in one European line's app is interesting because Keycloak is predominantly a self-hosted, enterprise-grade solution rather than a commercial SaaS offering. This suggests that whoever manages this line's identity infrastructure is running Keycloak on their own servers, likely to support a unified login experience that operates both onboard and shoreside without depending on a third-party cloud identity provider. The deep link scheme wiring in the app's manifest suggests Keycloak is being used for specific guest authentication flows, not just back-office access. As noted in the Security section, the presence of staging and test realm URLs in the production app's verified App Links configuration warrants a careful review of those environments.

LiteCore / Couchbase Lite

Couchbase — *couchbase.com*

Couchbase Lite is an embedded, document-oriented database with built-in sync capabilities. Its core engine appearing in one family-oriented line's app is a notable architectural signal. Couchbase Lite's standout feature is its ability to sync data between the device and a central Couchbase server when connectivity is available, then operate entirely offline when it is not. For a cruise app, this is a genuinely compelling architecture for handling anything from onboard activity bookings to dining menus to real-time folio data. Its deployment here suggests that team may be taking a more sophisticated approach to offline data management than most of the fleet.

Local WebSocket Push Service

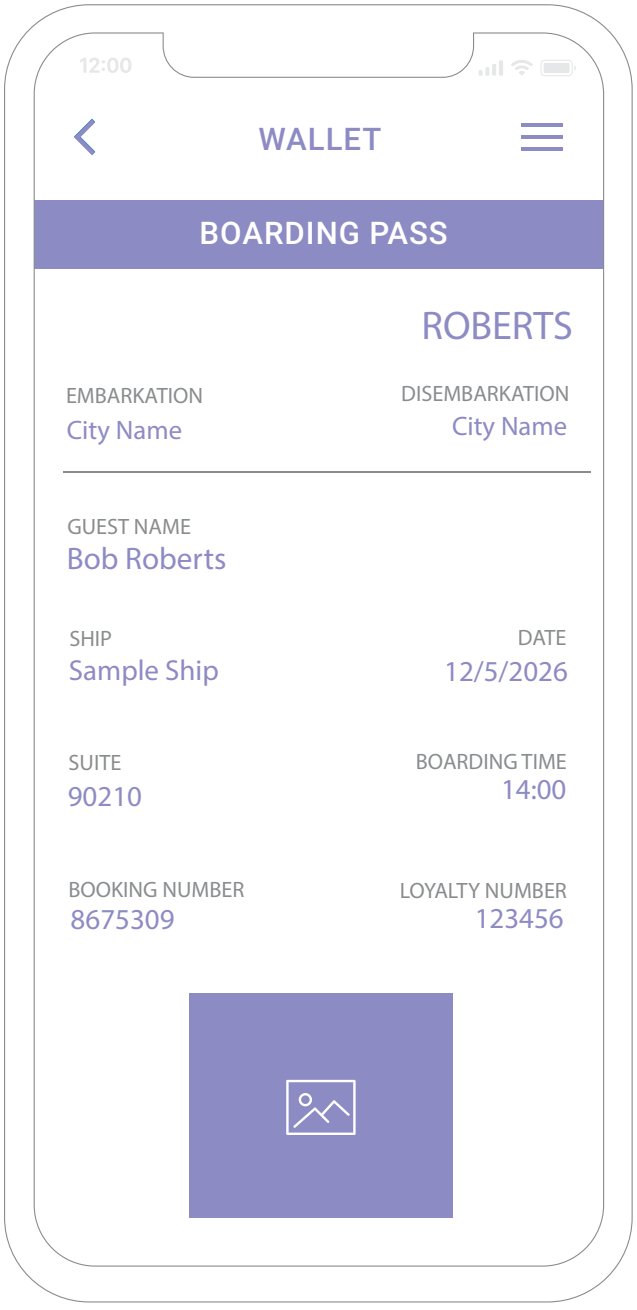
Various / Custom

One of the most practically significant patterns in the 2026 analysis is the proliferation of a persistent onboard WebSocket notification service. One app's manifest describes the intent most clearly: the foreground service is there to allow push notifications through a local websocket connection while the user is on the ship. This pattern, seen across four apps from different operators, represents a move away from the MQTT protocol that appeared in the 2024 report toward a more web-standard WebSocket model. The service is classified as foregroundServiceType=specialUse in Android, which requires explicit justification to Google during app review. These teams have clearly made a deliberate and documented architectural choice. This is likely the most significant shift in onboard notification infrastructure since the original local push implementations.

MapLibre

MapLibre / Open Source — *maplibre.org*

MapLibre is an open-source fork of Mapbox's original GL Native library, created after Mapbox relicensed its SDK away from open source in 2021. One operator's switch from Mapbox to MapLibre is the first we have seen in the cruise fleet and signals a growing preference for the license-free alternative among developers who want Mapbox-quality map rendering without the commercial dependency. If this trend continues, we may see more brands follow, particularly for bundled ship track and port map features where the tile data is hosted locally onboard and Mapbox's commercial value proposition is less obvious.



Notifee

Invertase / Open Source — notifee.app

Notifee is a React Native library for managing advanced local notifications, developed by Invertase, the same team behind React Native Firebase. It offers fine-grained control over notification channels, categories, and behaviors that the standard React Native notification APIs do not expose. Its inclusion in one major line's fully rebuilt React Native app makes good sense: the team was starting fresh after removing a legacy game engine dependency, and Notifee gives them full control over notification delivery as they build out a new onboard messaging architecture.

OwnID

OwnID — ownid.com

OwnID is a passwordless authentication platform that uses device biometrics to replace traditional password logins. It appeared independently in two apps that share a platform lineage and completed their cross-platform migrations in the same cycle. The fact that both brands added OwnID as part of the same migration is a strong signal that this was a deliberate platform-level choice rather than a coincidence. Passwordless login is particularly well-suited to cruise apps, where guests often download the app weeks before a voyage and then find themselves standing at the terminal with no

memory of what email address or password they used.

Pushy

Pushy LLC — *pushy.me*

Pushy is an alternative push notification service that operates outside of Google's Firebase Cloud Messaging ecosystem. One app's use of Pushy alongside a local notifications library suggests a belt-and-suspenders approach to notification delivery, possibly to handle edge cases where standard GCM delivery is unreliable. Pushy's documentation highlights its ability to work on devices where Google Play Services have been modified or restricted, which could matter in specific deployment or testing environments onboard ships.

React Native Blob Util

Open Source / RN community — *github.com*

React Native Blob Util is the community-maintained replacement for the now-deprecated rn-fetch-blob library. At least two React Native apps in the 2026 cohort document this swap explicitly in their dependency manifests — one as an upgrade, one adopting it from scratch in a full rebuild. This is a routine dependency modernization in the React Native ecosystem rather than a significant architectural shift, but its consistent appearance across multiple apps confirms that teams are actively keeping their file-handling utilities current.

React Native New Architecture

Meta — *reactnative.dev*

React Native's New Architecture replaces the old asynchronous bridge-based communication model with a JavaScript Interface (JSI) that allows direct synchronous calls between JavaScript and native code. The telltale signatures are landing across multiple cruise apps in 2026. The New Architecture generally improves performance for apps that pass large amounts of data between JavaScript and native code, which is common in feature-rich cruise apps with complex booking, itinerary, and folio interfaces. Apps that have not yet made this transition are increasingly in the minority.

Regula Document Reader

Regula Forensics — *regulaforensics.com*

Regula Forensics is a document verification company whose SDK is now appearing alongside Microblink in one large operator's app. Where Microblink focuses on fast, camera-based card capture, Regula brings heavier forensic document reading capabilities, including support for biometric chip reading from ePassports. Its inclusion suggests that this line's digital check-in process is being taken to a more rigorous verification level, possibly in line with port authority or customs requirements for specific itineraries. The combination of two document scanning SDKs in one app is unusual and points to a comprehensive identity verification strategy.

Rive

Rive Inc. — *rive.app*

Rive is a real-time interactive animation platform that allows designers to create animations driven by app state rather than playing back as pre-rendered clips. Its native library appearing in one brand known for unusually high design standards hints that their UI may incorporate animated elements that react dynamically to user input or application state. Rive

animations also tend to be significantly smaller in file size than equivalent video assets, which matters in an app already carrying an extensive SDK footprint.

Scanbot SDK

Scanbot — *scanbot.io*

Scanbot SDK is a commercial document and barcode scanning library with strong offline performance. Its appearance in one major line's rebuilt Flutter app is interesting because that same app also incorporates ML Kit barcode capabilities. Two scanning libraries in a single app is worth watching: the Scanbot SDK offers more advanced document capture features than ML Kit alone, including multi-page document scanning and data extraction, which may point to that line's digital check-in workflow receiving a significant upgrade as part of the broader rebuild.

SmartIdEngine

Smart Engines — *smartengines.com*

SmartIdEngine is a computer vision company specializing in ID document recognition. Their library appears in one major contemporary line's 2026 app as an updated version of a library present in their 2023 build. The naming change suggests a library update rather than a new adoption, but it is the first time SmartIdEngine has appeared prominently enough in this app to be worth calling out in the report. It is most likely being used as part of the passport or ID capture flow during the pre-cruise check-in process.

TWINT

TWINT AG — *twint.ch*

TWINT is a Swiss mobile payment scheme comparable to Apple Pay or Google Pay, but specific to the Swiss domestic market and widely used by both consumers and merchants there. Its presence in one premium line's app is a logical fit given that operator's strong Swiss and Italian customer base. This is one of the more geographically specific payment additions in this report and a good example of a cruise line tailoring its payment stack to the actual demographics of its guests rather than relying purely on universal providers.

Twilio Conversations

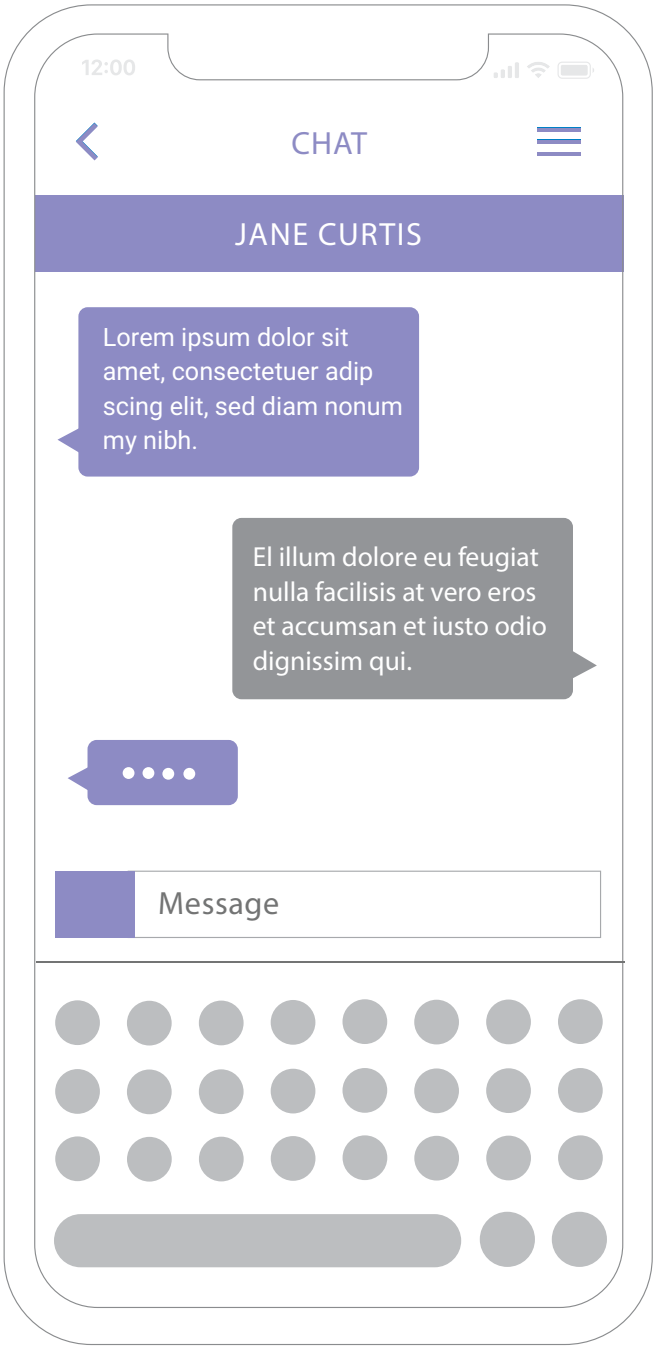
Twilio — *twilio.com*

Twilio Conversations is a messaging platform with support for text, media, and group channels. One family-oriented line's 2026 app includes both Twilio Conversations and Twilio Voice, together suggesting a fairly comprehensive in-app communication stack. Whether this is being used for guest-to-guest messaging, guest-to-crew communication, or both is difficult to determine from the binary alone. The combination of chat and voice from the same provider is a notable integration and likely reflects a deliberate decision to consolidate real-time communication under a single platform.

Twilio Voice

Twilio — twilio.com

Alongside Twilio Conversations, Twilio Voice enables in-app voice calling over the internet. Its presence alongside Twilio Conversations in one app is one of several signals in the 2026 report pointing to a broader expansion of voice calling capabilities into cruise apps. Given the challenges of shipboard cellular connectivity, an internet-based in-app calling feature could allow guests to connect with onboard services, shore excursion operators, or even family ashore when the ship's WiFi is available. It is one piece of what appears to be a significant communications infrastructure build.



UJET

UJET / Google Cloud — *ujet.cx*

UJET is a cloud-based contact center platform supporting both chat and voice interactions, and has been acquired by Google Cloud. Its concentrated presence across three apps belonging to the same major brand family is the most coordinated vendor adoption we observed in this report cycle. The 62-plus string key additions across two of those apps, combined with the third's full voice and chat service integration, suggest that this brand family has made a deliberate platform-level commitment to UJET for guest-facing support. The addition of chatbot deep link schemes in the same builds adds further evidence that this is connected to an AI-assisted support rollout, likely using Google's contact center AI capabilities that come bundled with the UJET platform.

VisionCamera

mrousavy/Open Source — *vision-camera.com*

VisionCamera is a React Native library for high-performance camera access that provides direct access to raw camera frames rather than just captured photos. It is commonly paired with computer vision workloads. Its inclusion in one expedition-style operator's app fits the broader picture of that build: a team investing in local map rendering, New Architecture performance improvements, and now more capable camera access. Exactly what they are using the camera frames for is unclear, but scanning and augmented reality applications are the most likely candidates.

Updated and Expanded

Updated

These are tools and platforms that appeared in the 2024 report and have meaningfully evolved in the 2026 cycle, whether through expanded adoption across more brands, significant version upgrades, or a notable shift in how they are being used.

Adyen

Adyen — adyen.com

Adyen first appeared in the 2024 report. In 2026, one app shows the addition of AdvancedCheckoutService and SessionCheckoutService, which reflect Adyen's newer API models for managing more complex checkout flows. This expansion suggests that the integration has matured beyond a simple card capture step into handling more of the transactional lifecycle directly within the app, including multi-step checkout sessions with fallback and retry logic.

Braze

Braze — braze.com

Braze's footprint has continued to grow, with one major entertainment-branded line showing expanded Braze activity alongside other marketing data tools. Braze is primarily used for shoreside engagement: pre-cruise welcome messages, upsell campaigns, loyalty program communications, and re-engagement targeting. Its increasing presence alongside newer data collection tools like the Adobe Experience Platform Mobile SDK suggests that some cruise lines are building more sophisticated marketing data pipelines than were visible in previous years.

Capacitor

Ionic — capacitorjs.com

Capacitor appeared in the 2024 report as an emerging Cordova replacement. In 2026, it has become the dominant migration story across one major brand family: two sibling lines completed full Cordova-to-Capacitor migrations in this cycle, and a third continued expanding its Capacitor plugin set. The shared codebase signatures between the two sibling lines strongly suggest these migrations were executed in parallel, possibly by the same team or vendor. Capacitor appears to have reached a maturity level where even large, feature-rich cruise apps are confident in completing the migration. One app in particular shows a well-developed Capacitor plugin ecosystem, including edge-to-edge display support, secure storage, and WebView overlay management.

ContentSquare

Contentsquare — contentsquare.com

ContentSquare continues to expand its presence across the fleet. In 2026, it reached one major line's fully rebuilt app, which is a meaningful data point: that team prioritized analytics instrumentation as a first-class concern in their rebuild rather than bolting it on afterward. ContentSquare provides session replay, heatmap, and funnel analytics for understanding how guests navigate complex multi-step flows like booking and check-in. For teams trying to optimize an experience that many guests only use once a year, having high-quality behavioral data is particularly valuable.

Cruise Director Platform

OnDeck by Sourcetoad — ondecksoftware.com

The Cruise Director Platform is OnDeck's suite of onboard and shoreside software tools covering middleware and API management, ship-to-shore data synchronization, push notifications, entertainment, digital signage, and guest-facing

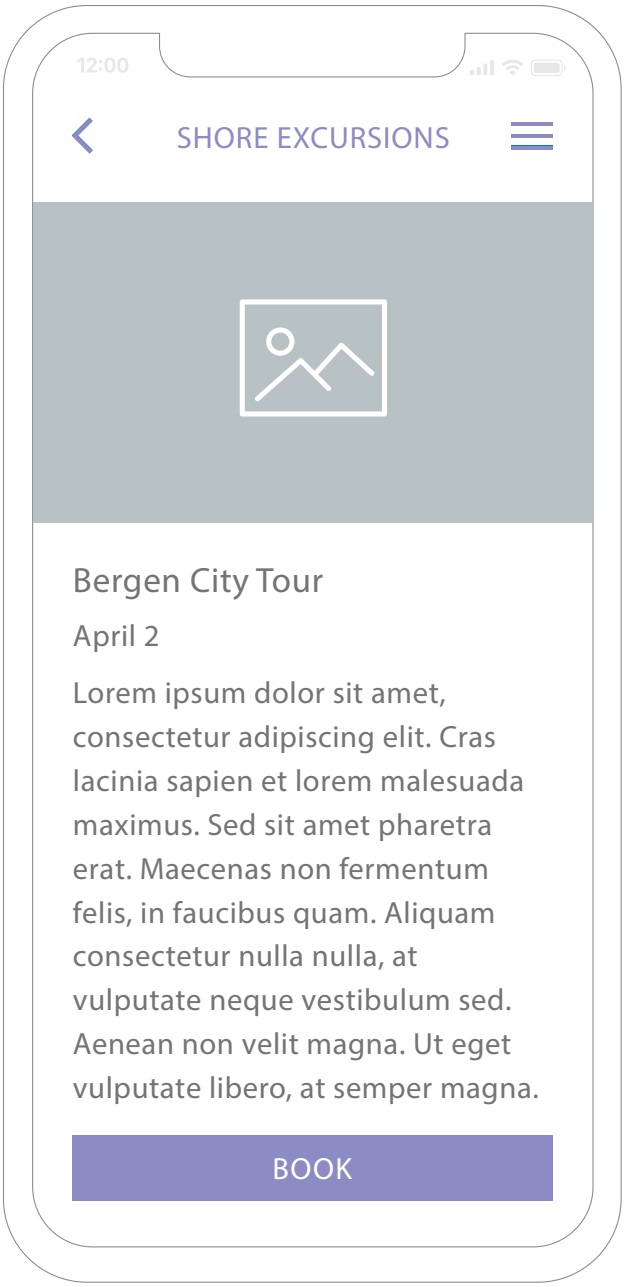
mobile app delivery. It is worth noting that OnDeck is the publisher of this report, and we have made every effort to analyze apps running on the Cruise Director Platform with the same methodology and standards applied to all other apps in the cohort.

What is analytically notable in the 2026 data is the range of operators running Cruise Director as their underlying platform. Four apps in this year's report share it as a common infrastructure layer: lines from different ownership structures, market positions, and geographic focuses. That kind of convergence across independent operators is a different signal from the brand-family consolidation we see elsewhere in the fleet. It suggests that the economics of building and maintaining proprietary app infrastructure are becoming harder to justify, and that purpose-built platform vendors are increasingly filling that gap across the industry.

Firebase

Google — *firebase.google.com*

Firebase remains the most ubiquitous backend platform in the fleet. In 2026, one notable update is a major version jump in the React Native Firebase wrapper across one large operator's app, reflecting significant changes across authentication, messaging, and analytics APIs. One signal worth interpreting carefully: a European-market line's 2026 build sets firebase_analytics_collection_enabled to false. It is tempting to read this as a brand reducing its analytics footprint, but that is probably the wrong conclusion. The same build expanded its consent SDK by nearly 190 string keys. The more likely explanation is consent-gating: Firebase analytics are disabled by default and only activated after the user grants consent through the CMP. This



is a compliant pattern for GDPR markets and is increasingly common in European-facing apps.

Flutter

Google — *flutter.dev*

Flutter continues to expand its footprint in the cruise fleet. One major contemporary line's complete native-to-Flutter rewrite is the biggest Flutter story in

2026, but the framework is also maturing in two other apps. That line's choice to rebuild entirely in Flutter rather than React Native is a meaningful data point in an ongoing industry debate. The two frameworks have different philosophies: Flutter's compiled rendering pipeline and stronger opinionation about UI give it different trade-offs than React Native's JavaScript bridge model, and that team has clearly decided those trade-offs favor Flutter for their use case.

Google MLKit

Google — *developers.google.com*

MLKit barcode scanning, which first appeared in the 2024 report, has expanded significantly in 2026 and now appears across several more apps. This consistent adoption is not surprising: ML Kit is fast, offline-capable, accurate, and maintained by Google. It has effectively become the default choice for barcode scanning in the fleet, replacing older solutions like ZXing. What is worth watching is whether these same apps will eventually expand their ML Kit usage into face detection or document recognition, both of which the SDK supports, as the check-in process becomes more biometrically sophisticated.

Hermes

Meta — *hermesengine.dev*

Hermes has taken another step forward in 2026. The appearance of libhermestooling.so as a separate library across multiple apps signals adoption of React Native 0.74 and above, which ships Hermes as the default engine with enhanced debugging and profiling tooling. Apps that have not yet made this transition are increasingly in the minority. The improved tooling in newer Hermes versions also handles New Architecture stack traces significantly better than older versions, which is becoming more important as the New Architecture rollout accelerates.

Mapbox

Mapbox — *mapbox.com*

Mapbox continues to be the dominant mapping solution in the fleet, though 2026 introduces a more complicated picture than previous years. One large operator significantly expanded its native Mapbox Maps SDK integration in a build that grew by nearly 100 megabytes. Meanwhile, another operator replaced Mapbox with the open-source MapLibre fork, the first such replacement we have seen in the fleet. Whether this represents the beginning of a cost-driven migration away from commercial Mapbox licensing or simply an isolated architectural choice will be worth watching in next year's analysis.

Microblink (BlinkID / BlinkCard)

Microblink — *microblink.com*

Microblink continues to be the fleet's primary solution for camera-based document and card capture. In 2026, its footprint expanded in two notable directions. One large operator added face biometric capabilities alongside the existing document scanning, and another major line adopted BlinkCard as part of their full React Native rebuild. The first deployment is particularly significant because it pairs Microblink with a forensic document reader, pointing toward a much more rigorous identity verification stack than a standard passport photo capture. This combination suggests that line's digital check-in process may be moving

toward compliance with more stringent port or customs verification requirements.

React Native

Meta — *reactnative.dev*

React Native remains the most widely deployed framework in the cruise app fleet. In 2026, the story is less about new adoption and more about maturation. The new architecture is rolling out across existing React Native apps, major dependency versions are being updated in step with the broader ecosystem, and several apps are completing long-overdue upgrades. One operator actually moved away from React Native back to native Android entirely. The framework appears to have reached a plateau where the interesting questions are no longer who is using React Native but how well they are using it.

Sentry

Functional Software — *sentry.io*

Sentry remains the most commonly identified crash reporting and error monitoring tool in the fleet. One large operator's build documents a significant version jump in the React Native Sentry wrapper, bringing improved performance tracing, session replay capabilities, and meaningfully better support for the New Architecture. For teams running complex React Native apps, keeping Sentry current is not purely cosmetic: newer versions handle New Architecture stack traces far more reliably than older ones, which becomes increasingly important as more apps complete that migration. As noted in the Security section, Sentry is also the tool most commonly associated with the sourcemap inclusion pattern — teams should ensure their Sentry configuration is scoped to upload sourcemaps to the Sentry service only, and not to bundle them within the distributable APK.

Declining and Removed New

Not everything moves forward. The 2026 cycle has a clear set of departures, some expected and some that arrived faster than anticipated. Taken together they tell a consistent story: the industry is consolidating around a smaller number of modern, maintainable foundations and shedding the legacy scaffolding that accumulated over the past decade.

ABTO VoIP SDK

ABTO Software — *abto.com*

ABTO's VoIP SDK was present in one app's previous build, providing SIP-based calling capabilities along with a notable native library footprint including multiple audio codec libraries. It was completely removed in that line's 2026 rebuild, along with the rest of its legacy stack. The rebuilt app appears to have moved away from in-app VoIP entirely, at least in this initial version. Given the broader industry move toward cloud-based contact center platforms, this removal may reflect a deliberate shift away from SIP-based telephony toward WebRTC or platform-managed voice solutions.

Airship

Airship — *airship.com*

Airship appeared as a new entrant in the 2024 report and has already been removed from one European line's 2026 build, replaced by the Adobe Experience Platform Mobile SDK. This swap is significant because it represents a deliberate

choice to consolidate marketing data collection under a single vendor rather than running separate notification and analytics platforms in parallel. Whether Airship's push notification capabilities have been fully absorbed by Adobe's tools or replaced by another system is not entirely clear from the binary alone, but the removal is notable given how recently it was added.

Cordova

Apache — *cordova.apache.org*

Cordova's decline continues in 2026. Two sibling lines within a major brand family completed full Cordova-to-Capacitor migrations in this cycle, leaving one smaller luxury operator as one of the last remaining Cordova holdouts in the fleet. That app's 2026 build shows no signs of migration planning. Cordova's tenure as a cross-platform foundation for cruise apps has been remarkably long, but the momentum has clearly shifted. Given the pace of change elsewhere in the fleet, it seems likely to follow eventually.

MQTT

OASIS Standard — *mqtt.org*

MQTT's role as the messaging backbone for onboard notifications appears to be ending. Its removal from one major contemporary line's app, combined with the proliferation of the WebSocket-based foreground service pattern across multiple brands, suggests that the industry is converging on a different approach for real-time onboard messaging. MQTT was a reasonable choice for its era and well-suited to constrained network environments, but WebSocket connections are more familiar to modern web developers and more compatible with the middleware stacks that cruise lines are building today.

Unity

Unity Technologies — *unity.com*

Unity's presence in cruise apps has been declining since the 2024 report. The most dramatic departure in 2026 was one large contemporary line whose complete rebuild moved away from both Unity and its previous cross-platform architecture, landing on a clean React Native codebase. One entertainment-branded line still includes Unity components, but the overall trajectory is clear. The maintenance complexity of combining a game engine with a consumer-facing mobile app has proved more challenging in practice than the initial promise suggested, and few new Unity adoptions are appearing in the fleet.

ZXing

Open Source — *github.com*

ZXing continues its decline in the fleet as Google MLKit barcode scanning gains wider adoption and matures. The 2024 report noted at least one app carrying unnecessary ZXing sample data that was inflating app size without adding any functionality. That pattern is improving as teams upgrade to ML Kit, which handles barcode scanning more efficiently and without the sample payload. ZXing is not disappearing overnight, but it is increasingly being treated as legacy infrastructure.

Contextual Mentions New

The following tools and signals are worth documenting, but carry less certainty than the entries above. Some are corroborated by only a single artifact, some are unattributed to a known SDK, and some are well-established tools that simply did not change enough in this cycle to warrant a full write-up.

Azure B2C / MSAL

Microsoft — *azure.microsoft.com*

Azure B2C continues to appear as a contextual signal in several apps, most clearly in one large contemporary line's build. Its role is likely as an enterprise identity layer for staff or partner logins rather than for guest-facing authentication, though the two are not mutually exclusive. That line's continued investment in Azure-based tooling is consistent with what we have seen in previous years. The addition of OwnID for passwordless guest authentication elsewhere in the fleet suggests that guest-facing identity is moving toward biometric solutions while enterprise identity stacks like Azure B2C hold steady.

Facebook Login SDK

Meta — *developers.facebook.com*

Facebook's login SDK appeared in one rebuilt app as a social authentication option. This is primarily used for guest logins via existing Facebook credentials, which remains a common shortcut in travel and hospitality apps for guests who prefer not to create a separate account. Whether this is a new feature or a capability carried over from the previous build is hard to confirm given the full package rewrite.

In-App Call Notification Strings

An identical cluster of seven call notification string keys appeared in three separate apps' 2026 builds. Per our analysis rules, string spikes are leads rather than conclusions and should be confirmed by supporting artifacts. In these three apps, no corresponding voice SDK native library was identified. The strings may originate from a shared framework dependency, a library that bundles call UI scaffolding speculatively, or a feature in early development that has not yet shipped a native component. We are noting it here as a signal worth watching rather than a confirmed capability. The same string pattern in three apps belonging to a single major brand family is corroborated by contact center platform service registrations and is reported under that entry.

OneTrust

OneTrust — *onetrust.com*

OneTrust remains a presence in several apps, providing GDPR and CCPA consent management infrastructure. Its continued appearance alongside newer entrants like Didomi is interesting: while Didomi appears to be the choice for one family of European-market apps, OneTrust holds its ground elsewhere in the fleet. The industry has not yet converged on a single platform for consent management, and the two products serve broadly similar functions for different brand families.

PDFium

Google — *pdfium.googlesource.com*

PDFium had a mixed 2026. One large contemporary

line removed it, while another major line added it back as part of a full React Native rebuild. This back-and-forth suggests that PDFium is used situationally for handling specific document types such as shore excursion tickets, onboard forms, or contracts in an offline environment, rather than as a core platform dependency. Its continued sporadic presence across the fleet is consistent with what we have seen in previous years.

QuickJS

Fabrice Bellard / Open Source — bellard.org

QuickJS is a lightweight, embeddable JavaScript engine created by Fabrice Bellard, best known as the author of FFmpeg and QEMU. Its appearance as libquickjs.so in one European expedition line's 2026 build is an unexpected find alongside the React Native runtime. QuickJS is sometimes bundled by third-party libraries as a sandboxed script execution environment, separate from the main JavaScript engine. Its exact purpose here is unclear, but it is worth noting that this app is among the more technically diverse in the fleet.

React-native-track-player

Doublesymmetry / Open Source — rntp.dev

react-native-track-player is a full-featured audio playback library for React Native. Its explicit presence in one expedition-style line's manifest via a registered MusicService foreground service tells us the app supports background audio playback, which could be used for anything from ambient onboard music channels to guided destination commentary. It is a logical fit: expedition-style voyages are content-rich experiences where audio guides for ports and destinations would be a natural feature.



Security Updated

If the 2024 report served as a wake-up call for the cruise industry on mobile app security, this year's findings suggest that the message has not yet been fully received. While some progress is evident, our analysis of the 2025 app cohort uncovered a cluster of findings that range from poor development hygiene to genuinely serious vulnerabilities — the kind that security teams at other software companies would treat as incidents requiring immediate remediation.

The cruise industry continues to present an attractive target for attackers. These apps handle passport data, payment credentials, and personally identifiable information for hundreds of thousands of passengers, and in many cases they maintain persistent connections to shipboard networks that are themselves connected to navigation and operational systems. The stakes are not abstract.

Development Configuration Shipped to Production

A persistent and underappreciated category of vulnerability is the leakage of development and staging configuration into production builds. We found multiple examples of this across the cohort this year.

One app ships its JavaScript bundle with a plain-HTTP development API endpoint embedded in the production build. HTTP endpoints — as opposed to HTTPS — are trivially interceptable and represent a meaningful man-in-the-middle risk if that endpoint is reachable at runtime, either because it is still active or because a future configuration change re-enables it. Beyond the immediate risk, the presence of a dev endpoint in a prod bundle is a reliable signal that build pipelines are not stripping environment-specific configuration before release — which raises the question of what else might be leaking.

A second app was found to include hardcoded references to non-production environments in its shipped strings, including clearly labeled development and non-production hostnames. While these strings may not be reachable at runtime under normal conditions, they expose the internal environment topology of the cruise line's infrastructure. In combination with other attack vectors — such as runtime feature flags or misconfigured staging systems that are still internet-accessible — leaked environment maps can meaningfully assist a targeted attacker.

Staging Realms in App Link Verification

One app's verified App Links configuration includes references to staging and test Keycloak authentication realm URLs. App Links are used to associate a mobile app with specific web domains for deep linking and authentication flows. Including staging realm URLs in a production app's verification manifest creates risk if those realms remain accessible and are not hardened to the same standard as production — which, in most organizations, they are not. A staging authentication realm that accepts real credentials, or that can be manipulated to issue tokens accepted by production systems, becomes a meaningful attack surface.

WebView Bridge Exposure

Another finding worth close attention is the addition of a WebView socket bridge in one app, introduced alongside a new Capacitor WebView overlay plugin. WebView-to-native bridges are a well-understood attack surface in mobile security: if the navigation allowlist is too permissive, if origin checks are absent or bypassable, or if the bridge exposes sensitive native APIs to untrusted web content, the result can be JavaScript injection into the native layer. We are not asserting exploitation here — the finding requires deeper dynamic analysis to fully characterize — but the pattern warrants a careful security review of the bridge's implementation before the next release.

Network Security Configuration Drift

One app appears to have removed its reference to an Android Network Security Configuration file from the application manifest, while the configuration file itself remains present in the APK. The practical effect depends on the content of that file: if it contained certificate pinning rules or restrictions on cleartext traffic, removing it from the manifest means those rules are no longer applied at runtime, even though they appear to still exist. This kind of configuration drift — where a security control is silently deactivated without being explicitly removed — is particularly dangerous because it may not be caught in functional testing. The app behaves identically; the protection is simply gone.

The Underlying Pattern

Taken together, these findings point to a consistent underlying problem that goes beyond any individual oversight: security is still not being treated as a first-class concern in the build and release process. Bundled credentials, dev endpoints in production bundles, staging configuration in verified manifests — these are not sophisticated vulnerabilities. They are the kinds of issues that a well-configured CI/CD pipeline with basic secret scanning and environment validation would catch automatically, before code ever reaches the Play Store.

The constructive path forward is not primarily about developers being more careful. It is about process. Automated secret detection tools — several of which are free and integrate directly into common CI systems — would catch a bundled .p12 file or a hardcoded dev endpoint before a build is promoted to production. Mandatory environment-specific build configurations, combined with post-build validation, would prevent staging hostnames from appearing in release artifacts. These are not exotic practices; they are standard in most mature software organizations.

The cruise industry serves a traveling public that is often far from home, on unfamiliar networks, handing sensitive personal data to apps they trust because they have to. That trust deserves a more rigorous defense than is currently evident in portions of the app cohort. The good news is that the tools and practices required to close these gaps are well understood and widely available. The investment required is primarily one of organizational will.

There is also a systemic complexity worth acknowledging. Even when cruise lines and their technology vendors want to improve security posture, the ecosystem of third-party integrations that surrounds a modern property management

system can make that genuinely difficult. We are aware of at least one case where full backend security controls were enabled across a fleet, only to be rolled back within hours because the number of third-party integrations that failed authentication was too large to address quickly. The controls were sound. The surrounding ecosystem was not ready for them. Fixing that requires coordination across vendors, not just internal will, and that coordination takes time. It is one more reason the "just fix it" framing undersells how hard this actually is in practice.

The Ostrich Problem

The technical findings in this section are concerning enough on their own. But there is a more fundamental issue underneath them that is harder to fix with a CI/CD pipeline.

In preparing this report, we spoke with a number of people across the industry about security practices. One exchange stood out. A senior technology executive at a large cruise company, when presented with specific vulnerabilities in their app, told us directly that they were not interested in hearing about them. Not because they disputed the findings, but because knowledge creates responsibility. If they do not formally know about a problem, they cannot be held accountable for failing to fix it. They were, in their own words, aware that things were wrong. They simply preferred not to have it confirmed.

We are not going to pretend this attitude is unique to the cruise industry. But it is worth naming plainly, because it explains something that pure technical analysis cannot: why the same categories of findings keep appearing year after year in apps maintained by organizations with the resources to fix them. The barrier is not capability. It is will.

The good news is that not everyone operates this way. We are aware of at least one major property management system vendor that recently went through a rigorous third-party security audit at the request of a cruise line customer, surfacing dozens of issues that were subsequently resolved. That is what responsible security practice looks like, and it is worth acknowledging alongside the less encouraging examples.

Rating our Predictions from 2024

A great deal has changed since our last full analysis of onboard ocean apps in 2024. Some trends materialize quickly, others take longer to develop, and a few fail to gain traction altogether. The 2024 report was written at a moment when several technologies were still emerging. The 2026 data shows which of those signals translated into meaningful implementation.

Enhanced Personalization

Personalization has advanced, but primarily at the infrastructure level rather than in the guest-facing experience. The expanded presence of platforms such as Adobe Experience Platform, Braze, and ContentSquare indicates that cruise lines are investing in data collection, segmentation, and behavioral analytics. Identity layers are also improving, enabling a more continuous relationship between guest and brand. However, most of this progress is upstream. The onboard experience itself remains relatively static, with limited evidence of real-time, context-aware personalization influencing itineraries, recommendations, or in-app interactions in a meaningful way.

Score: 6/10

Preparing for Chatbots

This prediction has largely materialized and, in some cases, moved beyond preparation into early execution. The 2026 data shows coordinated platform investments in AI-assisted guest services, including chatbot integrations and contact center platforms capable of handling both automated and human-assisted interactions. Early deployments are already handling meaningful service volume, indicating that these systems are no longer experimental. However, adoption is not yet universal, and capabilities still vary significantly between operators, keeping this just short of full realization.

Score: 7/10

Augmented Reality and Spatial Computing Features

While there are isolated technical signals pointing toward spatial capabilities, including advanced camera frameworks and indoor positioning technologies, there is little evidence of widespread or production-level adoption. AR remains confined to niche or experiential features rather than becoming a core part of the onboard app experience. The foundational components are appearing in select builds, but the broader ecosystem has not yet moved in a coordinated direction.

Score: 3/10

Improved Connectivity and Communication Tools

This prediction has been one of the most fully realized. The shift in onboard communication architecture is clear, with multiple apps adopting persistent WebSocket-based notification systems and expanding their messaging capabilities. At the same time, platform-level investments in chat, voice,

and contact center integrations signal a move toward more comprehensive communication ecosystems within the app. The groundwork for reliable, real-time onboard communication is now firmly in place, with only incremental improvements separating current implementations from full maturity.

Score: 9/10

Sustainability Tracking Features

We found no meaningful evidence of sustainability tracking features being implemented in the current cohort. Despite broader industry conversations around environmental impact and carbon reporting, these concerns have not translated into guest-facing functionality within mobile apps. This remains an area of unrealized potential.

Score: 0/10

Health and Safety Enhancements

Health and safety features, which saw rapid development during the pandemic years, have not continued to evolve in this cycle. The 2026 data does not show new or expanded functionality in this area, suggesting that development focus has shifted elsewhere. What remains in place appears to be baseline capability rather than an active area of innovation.

Score: 0/10

Enhanced Integration with Onshore Services

This prediction has fully materialized. The expansion of pre-cruise functionality, including booking, identity verification, and shoreside engagement, reflects a clear shift toward treating the mobile app as a continuous travel companion rather than an onboard-only tool. The technical foundations required to support this integration, including improved authentication, data synchronization, and backend connectivity, are now in place across multiple operators. This is no longer an emerging trend but a defining characteristic of leading apps in the fleet.

Score: 10/10

Advanced Analytics for Personalized Marketing

The analytics ecosystem has matured significantly since the last report. The coordinated adoption of modern analytics and engagement platforms, combined with deeper instrumentation in newly rebuilt apps, demonstrates a clear investment in understanding and influencing guest behavior. These systems are now capable of supporting sophisticated marketing and engagement strategies across the full guest lifecycle. While execution will vary by operator, the underlying capability is firmly established.

Score: 8/10

Looking Ahead

If the 2026 analysis tells a story of consolidation and maturation, the next two years look like execution. The foundations have been laid: frameworks modernized, notification infrastructure rebuilt, identity stacks upgraded. What follows is our best read on where the industry goes next.

Offline-First Booking Will Arrive, Unevenly

One of the more technically ambitious shifts on the horizon is the ability for guests to make onboard bookings from their mobile device while off the ship. Dining reservations, shore excursions, spa appointments. The demand is obvious: guests on a four-hour excursion want to secure dinner before they return to the pier, not queue at guest services when they get back.

The challenge is equally obvious. Making a reservation on a ship requires talking to a property management system that lives on that ship. When the guest is ashore, the app needs an internet connection that routes through a shoreside proxy, which then communicates down to the vessel. That is not an insurmountable architecture problem, but it requires reliable shoreside infrastructure, careful handling of conflict states when connectivity drops mid-transaction, and enough confidence in the data sync model to commit a booking against a live inventory system the app cannot directly reach.

We have seen evidence across at least three or four apps of teams working through exactly these problems, and they are using meaningfully different approaches to get there. Some are investing in embedded databases with sync capabilities. Others are building WebSocket-based service layers that persist through intermittent connections. None of them appear to have fully shipped the feature yet, but the groundwork is visible. Expect the first production implementations to appear in the next cycle, and expect them to be imperfect. The edge cases in this problem space are genuinely hard, and the first versions will likely be limited to lower-stakes bookings before expanding to more complex transactions.

Pre-Cruise Experiences Will Continue to Deepen

The pre-cruise portion of the app experience has been growing steadily for several years, and the 2026 data suggests that trajectory is accelerating. Digital check-in has become table stakes. The next wave appears to be a more complete pre-voyage concierge experience: activity booking, dining reservations, itinerary planning, and personalized recommendations all available weeks before the guest sets foot on the ship.

The identity verification infrastructure being built right now directly enables this. Passwordless login through biometrics, ePassport reading, and face-matching are not just faster check-in tools. They are the foundation for a continuous authenticated relationship between the guest and the cruise line from the moment of booking through post-voyage follow-up. Expect the pre-cruise window to

get longer and more feature-rich, and expect the boarding process itself to get meaningfully faster as a result.

At least two operators have confirmed a pre-cruise secondary booking feature in active development, targeting a rollout in early 2026. The architecture required to make this work is non-trivial: shoreside booking requests need to route through an internet-accessible proxy that communicates down to the property management system on the vessel, with enough reliability and conflict-handling to commit transactions against a live inventory system the app cannot directly reach. The fact that this is now in active development rather than on a roadmap is a meaningful signal. Expect others to follow once the first implementations demonstrate it is achievable.

Biometric Check-In Will Become Expected

The combination of passwordless login, face biometric capabilities, ePassport chip reading, and expanded document scanning all expanding in the same cycle is not a coincidence. The industry is visibly assembling the components of a check-in experience that is faster, more secure, and less dependent on physical documents than what exists today. The individual pieces are present across multiple apps right now. The prediction is that they start getting connected into coherent end-to-end flows, and that by the next report cycle at least two or three brands will have a demonstrably faster embarkation process to point to as a real competitive differentiator.

In-App Voice Is Being Wired In

Several signals in the 2026 data point toward in-app voice calling becoming a first-class feature. A major brand family has made a coordinated platform commitment to a cloud contact center service that includes both chat and voice. A second operator has integrated both a chat and a voice SDK from the same provider. A third cluster of apps shows identical call notification string patterns without yet shipping a corresponding native library, which is a pattern we have seen before with features in late development.

The use cases are varied: guest-to-crew service requests, AI-assisted support that can escalate to a live agent, in-app calling to shore excursion operators, and potentially guest-to-guest communication onboard. Not all of these will ship in the next cycle, but at least one brand offering functional in-app voice calling as a supported feature seems likely.

AI-Assisted Guest Services Will Move From Experiment to Expectation

The 2024 report noted the first signs of AI-assisted chat in cruise apps. The 2026 data suggests that transition is accelerating. A coordinated rollout of a Google Cloud-backed contact center platform across a major brand family, combined with chatbot deep link schemes already wired into production builds, points to functional AI-assisted guest service appearing in at least a few apps in the next cycle.

The more interesting question is not whether it works, but whether it handles meaningful transaction volume. The industry's previous experiments with virtual assistants often stalled at novelty, capable of answering simple FAQs but unable to complete actions. The new generation of contact

center AI, backed by large language models and integrated directly into booking and service APIs, is meaningfully more capable. We are aware of at least one operator that has reported handling over 13,000 service requests through a virtual concierge in a six-month period across a portion of their fleet. That is not a pilot number. That is operational scale, and it points to something more significant than a chatbot answering questions about pool hours.

What is also worth noting is where that volume is coming from. The assumption tends to be that AI concierge features will be used primarily for dining reservations and excursion bookings. In practice, a significant share of the demand appears to be for basic service requests: extra pillows, housekeeping timing, bed configuration, towel delivery. These are requests that currently require a phone call or a trip to guest services, and they are exactly the kind of low-stakes, high-frequency interactions that a well-integrated AI assistant handles well. The first brands to connect this properly to their housekeeping and service management systems will have a genuine operational advantage, and the guest experience benefit will be immediately visible.

Compliance Tooling Will Consolidate Around a Small Number of Serious Contenders

The consent management picture in 2026 is not one of convergence on a single platform, but it is beginning to look like a two-horse race. OneTrust and Didomi are both present in the fleet, serving different brand families, and both are expanding their implementations rather than being replaced. What is less clear is whether the remaining apps that handle consent through simpler mechanisms, basic permission dialogs and home-built solutions, will eventually migrate to one of these platforms as regulatory scrutiny increases.

The prediction is not that the fleet standardizes on a single CMP, but that the number of credible options narrows. As GDPR enforcement matures and similar frameworks spread to other markets, the cost of a home-built or minimal consent solution will go up. The brands that have already invested in a serious CMP will be better positioned, and the laggards will increasingly find that the build-your-own approach is not worth the compliance risk.

Security Hygiene Will Improve, Slowly

The findings documented in the Security section of this report are not new categories of vulnerability. They are well-understood problems with well-understood solutions. The optimistic reading of the 2026 data is that the industry is aware of the gap and beginning to close it: bot management tooling is being embedded at the native level, rooted device detection is present in several apps, and certificate management is receiving more attention than it was two years ago.

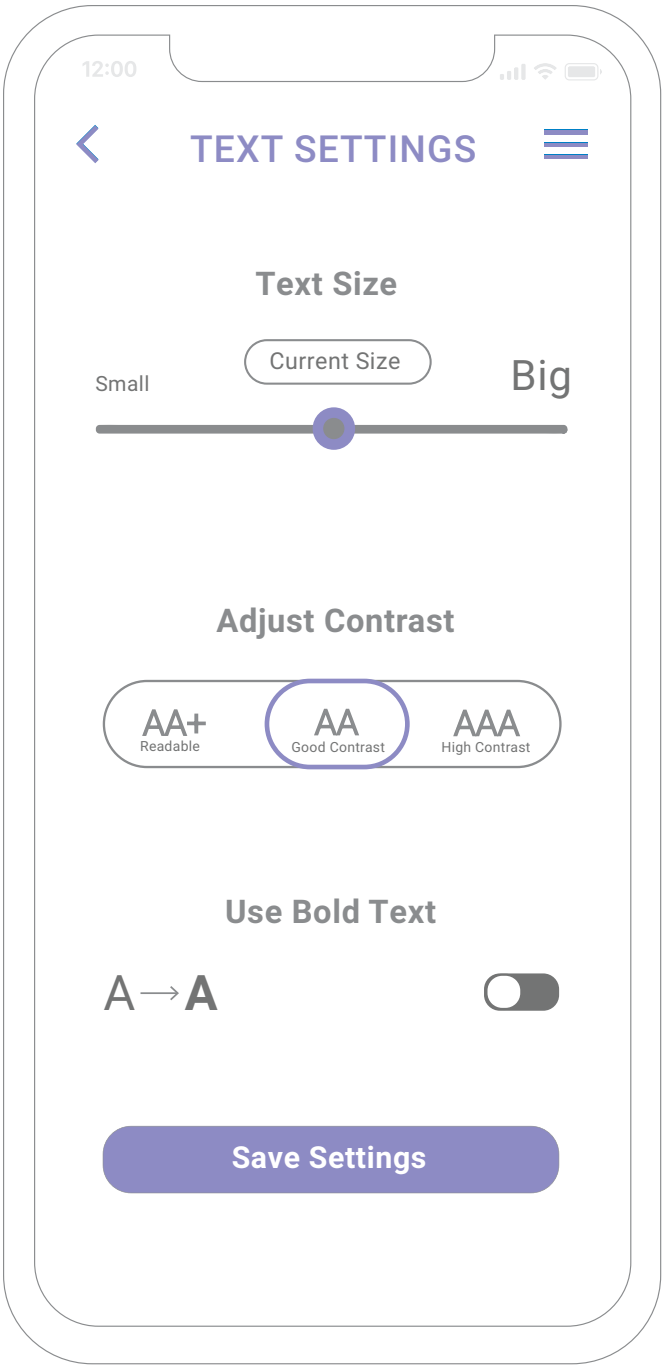
The less optimistic reading is that the most serious findings, bundled credentials, development configuration in production builds, staging endpoints in verified manifests, represent gaps that automated tooling would catch before a build ever reached the Play Store. Until those build pipeline controls are standard practice across the fleet, these findings will recur. The prediction is that at least one or two brands make meaningful improvements in this area

following the publication of this report, but that the industry as a whole will still have significant ground to cover in the 2027 cycle.

Accessibility Will Receive More Attention, But Progress Will Remain Uneven

The accessibility picture in cruise apps has been improving, but from a low baseline. The addition of screen reader support, high-contrast modes, and media accessibility features in a handful of apps is encouraging. What is less encouraging is that these improvements are concentrated in a small number of apps while the majority of the fleet has no visible accessibility strategy.

The forces pushing this forward are real. ADA litigation risk is rising, WCAG compliance is increasingly expected by regulators and institutional travel buyers, and the cruise industry's passenger demographics skew older, a population with higher rates of visual, auditory, and mobility challenges than the general public. Expect to see more apps adding at minimum the baseline accessibility features that have become standard elsewhere in travel and hospitality. Expect the leading apps to go further, and expect the gap between the leaders and the laggards to become more visible and more legally consequential over the next two years.



Conclusion

Every year we write a conclusion and every year we try to resist the temptation to say that this was the most important year yet. But 2026 is going to be difficult to top.

The structural story of this report is consolidation. The cross-platform framework debates that defined the last several years are settling into something more predictable. React Native is maturing rather than expanding. Flutter has made its case and won it in at least one significant rebuild. Capacitor has effectively replaced Cordova across a major brand family, and nobody is arguing about whether that was the right call. The experimental phase of cruise app development, at least at the framework level, is largely over. What replaces it is harder and more interesting: the question of whether these teams can execute well at scale, ship features that actually work onboard, and close the security and accessibility gaps that have followed this industry for years.

The AI story is where things get genuinely exciting. Two years ago we predicted that AI-assisted chat would move from experiment to early production. It has. And it has moved faster and into more interesting territory than we expected. The 13,000 service requests handled by a single operator's virtual concierge in six months is the kind of number that changes conversations in boardrooms. It is not just a feature metric. It is an operational one, and it points toward a future where the ratio of crew to guests looks different because the low-stakes, high-frequency interactions are being handled by software.

The pre-cruise experience is the other area we are watching closely. The pieces are assembling: biometric check-in, passwordless authentication, ePassport reading, pre-cruise secondary booking in active development. What has been a fragmented set of features is starting to look like a coherent strategy at the leading end of the fleet. Embarkation is going to get meaningfully faster for guests at lines that have been making these investments, and that speed difference will be noticeable enough to become a competitive argument.

What has not improved as fast as it should is security. We understand the ecosystem complexity. We have documented the vendor coordination challenges. We have acknowledged that fixing these problems is harder than it looks from the outside. But we also named the other thing this year, the organizational culture that prefers ignorance to accountability, and we are not going to soften that message in the conclusion. The apps in this report are trusted by hundreds of thousands of passengers with their passport data, their payment credentials, and in some cases their physical location. That trust is not unconditional, and it should not be treated as though it is.

The good news is that the tools, the practices, and increasingly the will to do better are all present in this industry. We have seen rigorous security audits carried out at the request of cruise line customers. We have seen development teams making sophisticated architectural choices for exactly the right reasons. We have seen apps that are genuinely

excellent. The question for the 2027 cycle is whether the rest of the fleet closes the gap, or whether the distance between the leaders and the laggards continues to grow.

We will be watching, and we will be writing about it. Thank you for reading.