Braze iOS SDK Certification Practice Test (Sample)

Study Guide

Everything you need from our exam experts!

Questions

1. What is the default interval for data flushing to Braze?

A. 5 seconds

B. 10 seconds

C. 15 seconds

D. 30 seconds

2. What purpose do custom events serve in Braze?

A. To track user progress

B. To enhance app design

C. To facilitate user authentication

D. To manage content delivery

3. True or False: Data points are counted for both session start and session end.

A. True

B. False

C. Depends on configuration

D. Not applicable

4. Which method checks if an app was launched by a Braze internal push notification?

A. App.Launch.isInternalLaunch

B. Braze.Notifications.isInternalNotification

C. Notifications.Check.isInternal

D. Braze.Launch.isNotified

5. What action is NOT typically required when troubleshooting a web push issue?

A. Checking if foreground push is enabled

B. Verifying browser and OS-level notifications are allowed

C. Ensuring the test user is familiar with push notifications

D. Using verbose logs for investigation

6. When a user returns to their session after closing their tab, what will they experience if they return within the timeout period?

A. A new session

B. They will not be logged in

C. The same session

D. Session expiration

7. Which method can be used to limit automatically collected data sent to Braze?

A. Data Redaction

B. Using an allowlist

C. Removing data manually

D. Compression of data

8. What does APNs stand for?

A. Apple Push notification service

B. Application Programming notification service

C. Android Push Notification system

D. Application Push network system

9. Where are push tokens stored in Braze?

A. On the user's device

B. On the user profile

C. In the notification center

D. In the app settings

10. In Braze, how are user sessions typically maintained?

A. Through push notifications

B. By user authentication

C. Session tokens

D. Cookies

Answers

1. B
2. A
3. A
4. B
5. C
6. C
7. B
8. A
9. B
10. C

Explanations

1. What is the default interval for data flushing to Braze?

A. 5 seconds

B. 10 seconds

C. 15 seconds

D. 30 seconds

The default interval for data flushing to Braze is indeed 10 seconds. This setting is crucial for optimizing the user experience and ensuring that data generated by the app is sent to Braze promptly for real-time engagement and analytics. When the system flushes data at this interval, it strikes a balance between reducing the frequency of network calls and ensuring timely updates. By sending data every 10 seconds, the app can maintain a fresh dataset on user interactions and behaviors, enabling personalized marketing efforts more effectively. In this context, a shorter interval may lead to excessive network requests, which can drain device resources and potentially hinder performance. Conversely, a longer interval might delay the availability of critical user data for engagement campaigns. Thus, the 10-second default interval is designed to provide a reliable and efficient mechanism for data synchronization with the Braze platform.

2. What purpose do custom events serve in Braze?

A. To track user progress

B. To enhance app design

C. To facilitate user authentication

D. To manage content delivery

Custom events in Braze are designed primarily to track user interactions and progress within an application. By defining custom events, developers can log specific actions taken by users, such as completing a purchase, reaching a certain level in a game, or using a specific feature. This data is invaluable for analyzing user behavior and understanding how users are engaging with the app. Tracking user progress through custom events allows businesses to gain insights into user journey patterns and identify areas for improvement in user experience or engagement strategies. This analytical capability can drive personalized marketing efforts, enhance user retention strategies, and ultimately contribute to better overall app performance. Understanding how users interact with different features or content is essential for making data-driven decisions that can optimize the app and meet user needs more effectively.

3. True or False: Data points are counted for both session start and session end.

A. True

B. False

C. Depends on configuration

D. Not applicable

The correct answer is true because in the context of data tracking and analytics, both the start and end of a session are important events that can provide valuable insights. When a session starts, it marks the beginning of a user interaction period with the app, allowing for the collection of metrics related to user engagement, duration of the session, and other interaction events. Similarly, marking the end of a session allows analysts to capture data about how users were interacting with the app leading up to that point. This dual data collection helps in understanding user behavior patterns, identifying potential issues within the app, and optimizing user experience. In many analytics frameworks, including Braze, it is standard to log both session start and session end as data points to provide a comprehensive view of user activities. This practice helps businesses and developers to track engagement accurately and make informed decisions based on user behavior trends.

4. Which method checks if an app was launched by a Braze internal push notification?

A. App.Launch.isInternalLaunch

B. Braze.Notifications.isInternalNotification

C. Notifications.Check.isInternal

D. Braze.Launch.isNotified

The method that checks if an app was launched by a Braze internal push notification is correctly identified as Braze.Notifications.isInternalNotification. This method is designed to determine the source of the app launch, specifically for cases where the launch was triggered by a Braze push notification. Utilizing this function allows developers to implement conditional logic based on whether the app was opened through Braze's internal notifications. This capability is essential for optimizing user experience and engagement strategies that are associated with notifications, enabling targeted content delivery and actions in the app after determining the onset of user interaction via such notifications. The other methods listed are not specifically designed to serve this purpose, which confirms the selection of the correct answer.

5. What action is NOT typically required when troubleshooting a web push issue?

A. Checking if foreground push is enabled

B. Verifying browser and OS-level notifications are allowed

C. Ensuring the test user is familiar with push notifications

D. Using verbose logs for investigation

When troubleshooting a web push issue, understanding user familiarity with push notifications is not typically necessary for resolving technical problems. The focus in troubleshooting should be on the technical configurations and platform settings that directly affect the ability to send and receive notifications. Checking if foreground push is enabled, verifying that browser and OS-level notifications are allowed, and using verbose logs for investigation are all practical and technical steps that directly pertain to ensuring that the push notifications are functioning correctly. These actions involve examining the environment set up and the application's settings to diagnose any issues, while the user's understanding does not impact the technical operation of the push system itself.

6. When a user returns to their session after closing their tab, what will they experience if they return within the timeout period?

A. A new session

B. They will not be logged in

C. The same session

D. Session expiration

When a user returns to their session after previously closing their tab, if they do so within the specified timeout period, they will experience the same session. This means that all the information, preferences, and states that were active during their initial session will be preserved and accessible. This continuity is designed to create a seamless user experience, allowing users to pick up where they left off without needing to re-authenticate or lose their context. The timeout period is a critical factor in this scenario. It defines the duration that the session remains active after last use. If the user returns before this period ends, the session is still recognized, and they can continue using the application as if they never left. This functionality enhances user retention and engagement by minimizing disruptions and allowing users to navigate the app more fluidly, thereby encouraging longer interactions and satisfaction with the app's functionality.

7. Which method can be used to limit automatically collected data sent to Braze?

A. Data Redaction

B. Using an allowlist

C. Removing data manually

D. Compression of data

The correct method to limit automatically collected data sent to Braze is through the use of an allowlist. An allowlist enables developers to specify which data points should be collected and sent to Braze, ensuring that only desired data is transmitted. This approach allows for a more tailored data collection strategy, helping organizations manage privacy concerns and comply with regulatory requirements by filtering out any data they do not want to share. In contrast, data redaction typically involves editing or removing certain pieces of information from data records, which is more related to managing existing data rather than limiting what is collected initially. Removing data manually can be cumbersome and may not provide the proactive control that an allowlist offers. Compression of data, while useful for reducing the size of data being transmitted, does not inherently limit the type of data collected, but rather focuses on optimizing the transmission process. Thus, using an allowlist is the most effective method for controlling the collection of automatically generated data.

8. What does APNs stand for?

A. Apple Push notification service

B. Application Programming notification service

C. Android Push Notification system

D. Application Push network system

APNs stands for Apple Push Notification service. This is a service provided by Apple that enables developers to send push notifications to iOS devices. Such notifications can include messages, alerts, badges, and more, allowing apps to engage and inform users even when the app is not actively being used. This terminology is crucial for developers working with the Braze iOS SDK, as push notifications are a significant part of how Braze engages users through messaging and notifications. Understanding the function and definition of APNs is essential for integrating push notifications effectively in iOS applications. The other options presented do not accurately reflect the service provided by Apple. "Application Programming notification service" misrepresents the focus of APNs as it is not a programming interface. "Android Push Notification system" incorrectly refers to a service meant for Android devices rather than Apple. Finally, "Application Push network system" is a generic name that does not correspond to the specific functionality and purpose of Apple's service.

9. Where are push tokens stored in Braze?

A. On the user's device

B. On the user profile

C. In the notification center

D. In the app settings

The correct answer is that push tokens are stored on the user profile within Braze. When a user installs an app that integrates with Braze and grants permission for push notifications, the SDK generates a unique push token that identifies the device to which messages can be sent. This push token is crucial for targeting specific users with notifications and personalized content. Storing the push token on the user profile ensures that it is associated with the specific user and their activity within the app. This allows for efficient management of notifications and campaigns targeting that user. Additionally, if the user reinstalls the app or changes devices, the token can be updated and managed appropriately within their user profile, maintaining the integrity of the messaging system. In contrast, storing the push token on the user's device would limit accessibility for the Braze platform to run analyses and manage notifications effectively. Similarly, the notification center is not designed to manage tokens but rather serves as an interface for displaying notifications. Storing push tokens in the app settings would also be inefficient, as that information needs to be linked with user data for effective targeting and audience segmentation.

10. In Braze, how are user sessions typically maintained?

A. Through push notifications

B. By user authentication

C. Session tokens

D. Cookies

User sessions in Braze are typically maintained using session tokens. Session tokens are unique identifiers that are generated when a user starts a session within your app. These tokens help the SDK recognize and differentiate between individual user sessions, allowing for seamless tracking of user interactions and behaviors over time. When a user interacts with the app, the session token is used to associate their actions with their profile in the Braze platform. This is crucial for personalizing user experiences and analyzing engagement metrics, as it helps the Braze system maintain continuity of the user's journey, even if they navigate away from the app and come back later. Other methods, such as push notifications, user authentication, and cookies, do not serve the same primary purpose within the Braze ecosystem of maintaining user sessions. While they can play roles in user engagement and interactions, they are not specifically designed for session management like session tokens are. This highlights why session tokens are the correct answer for maintaining user sessions in Braze.