Below is a detailed Solution Design Reference (SDR) for the implementation of Adobe Analytics on the two provided Green Pay mobile app screens using Adobe Launch, the Adobe Experience Platform (AEP) Mobile SDK, and Adobe Analytics. The goal is to capture key user interactions and screen views, align them with analytics dimensions and metrics, and have the data flow into both Adobe Analytics and Adobe Experience Platform.

Table of Contents

1. Overview and Goals

Primary Objective:

Implement robust tracking on the Green Pay mobile app's login and dashboard screens using the Adobe Experience Platform Mobile SDK and Adobe Launch. The captured data should flow into Adobe Analytics for reporting and also be available in the Adobe Experience Platform (AEP).

Key Use Cases:

- Understand how often users land on the login screen and what actions they take (e.g., login attempts, forgot password, create account).
- Measure login success/failure rates.
- On the dashboard screen, understand user engagement with features (e.g., "Create New…" action, "Scan Document") and track metrics related to account status (e.g., number of overdue invoices, upcoming payouts).

Benefits:

- Optimize the login funnel to increase successful logins and reduce friction points.
- Gain insights into which features users engage with most on the dashboard.
- Leverage this data within AEP for advanced segmentation, personalization, and activation.

2. Technical Architecture

Key Components:

- **Adobe Launch:** Tag management to configure rules, data elements, and orchestrate data collection.
- **Adobe Experience Platform (AEP) Mobile SDK:** Integrates with Launch via the Edge Network for data collection.
- **Adobe Analytics:** Destination for behavioral analytics and reporting.
- **Adobe Experience Platform:** Unified data store for real-time customer profile and analysis.

Data Flow:

1. User opens the app and navigates to a given screen or interacts with UI elements.
2. The Mobile SDK sends data (trackState for screens, trackAction for interactions) to Adobe Launch.
3. Launch rules process the data, map variables, and send it to Adobe Analytics and AEP via Experience Event Forwarding.
4. Data is available in Adobe Analytics for reporting and in AEP for advanced use cases.

3. Data Collection Methodology

3.1 Screen-Level Tracking

Use trackState() calls on each screen load.

- **Login Screen:**
  - Screen Name: login_screen
  - Trigger: On screen view.
- **Dashboard Screen:**
  - Screen Name: dashboard_screen
  - Trigger: On screen view after the user successfully logs in.

3.2 Interaction Tracking

Use trackAction() calls for user interactions such as button taps.

- **Login Screen Interactions:**
  - "Login" button tap
  - "Forgot Password?" link tap
  - "Create your Account" link tap
  - Login Success/Failure (after user attempts login)
- **Dashboard Screen Interactions:**
  - "Create New…" button tap
  - "Scan document" button tap
  - Potential future interactions (e.g., tapping an invoice card)

3.3 User Identification and Context Data

- Once a user logs in, capture a hashed user ID or account ID for persistency.
- Include contextual information such as user segment (if available), timestamp, app version, and device type.
- The user's organization name and greeting message (e.g., "John Doe Entertainment Inc." and user's first name "John") can be captured as context to enrich the data.

4. Variables and Event Definitions

4.1 eVars (Persistent Custom Dimensions)

eVar #	Name	Description	Expiration	Example Value
eVar1	Screen Name	Captures the name of the current screen	On Page View	login_screen, dashboard_screen
eVar2	User ID (Hashed)	Unique identifier for logged-in users	Visit	hash_abc123
eVar3	Login Method	Method used to login (if applicable)	Hit	username_password
eVar4	Account/Business Name	Name of user's business account after login	Visit	John Doe Entertainment Inc.

4.2 Props (Non-Persistent Custom Dimensions)

prop #	Name	Description	Example Value
prop1	App Section	Captures high-level section of the app	login, dashboard
prop2	Button/Link Clicked	Captures the label of the clicked element	login_button, forgot_password
prop3	Month (Contextual)	Month when the event occurred (for trending)	January, February
prop4	CO2 Reduction Graph Month	Month associated with carbon footprint graph	Jan, Feb (from chart)

Note: prop4 may be set dynamically if needed when user interacts with the graph or when the dashboard loads.

4.3 Events (Custom Conversion Events)

Event #	Name	Type	Description
event1	Screen View	Counter	Fires when a screen is viewed
event2	Login Attempt	Counter	Fires when user attempts to log in
event3	Login Success	Counter	Fires upon successful login
event4	Login Failure	Counter	Fires upon failed login attempt
event5	Feature Interaction	Counter	Fires when user taps on a feature button/link
event6	Overdue Invoices Shown	Counter	Fires when dashboard loads showing overdue invoices
event7	Upcoming Payouts Shown	Counter	Fires when dashboard loads showing upcoming payouts
event8	Instant Payout Shown	Counter	Fires when dashboard loads showing instant payout info

Note: Dashboard metrics such as “4 overdue invoices” or “$225,110.30” could be captured as context data and possibly mapped to Analytics variables or AEP XDM fields. If required, numeric events can be created to track numeric values (e.g., event9: Overdue Invoice Count set to 4).

4.4 Context Data Variables

Context data will be used as a staging area in Launch. These keys will be set in the app code and mapped in Launch to corresponding eVars, props, and events.

Example Context Data Keys:

- context.screen.name = login_screen / dashboard_screen
- context.user.id_hashed = hash_abc123
- context.user.account_name = John Doe Entertainment Inc.
- context.login.attempt = true (or false)
- context.login.status = success / failure
- context.button.clicked = login_button, forgot_password, create_account, create_new, scan_document
- context.invoice.overdue_count = 4
- context.invoice.overdue_amount = 225110.30
- context.invoice.upcoming_payouts = 112333.45
- context.invoice.instant_payout = 50010.30

These context data values will be mapped to Analytics variables as needed.

4.5 XDM Mapping for Adobe Experience Platform

Data sent to AEP should conform to an XDM Experience Event schema. Sample mappings:

- XDM Field: xdm._experience.pageDetails.pageName <- context.screen.name
- XDM Field: xdm.customerAccount.id <- context.user.id_hashed
- XDM Field: xdm.customerAccount.name <- context.user.account_name
- XDM Field: xdm.commerce.interaction.type <- context.button.clicked
- XDM Field: xdm.commerce.payout.overdueCount <- context.invoice.overdue_count
- XDM Field: xdm.commerce.payout.overdueAmount <- context.invoice.overdue_amount

Additionally, the login success/failure can be represented as separate XDM event types or with a dimension like xdm.eventType = "loginAttempt" and xdm.eventStatus = "success" or "failure".

5. Data Layer and Tag Management Configuration (Adobe Launch)

5.1 Rules

Example Rules:

1. **Rule: Page Load - Login Screen**
   - **Event:** trackState call with context.screen.name = login_screen
   - **Conditions:** None
   - **Actions:**
     - Set eVar1 = login_screen
     - Set prop1 = login
     - Set event1 (Screen View)
2. **Rule: Page Load - Dashboard Screen**
   - **Event:** trackState call with context.screen.name = dashboard_screen
   - **Conditions:** User is logged in (context.user.id_hashed present)
   - **Actions:**
     - Set eVar1 = dashboard_screen
     - Set prop1 = dashboard
     - Set event1 (Screen View)
     - Set event6, event7, event8 if applicable based on invoice data availability

3. **Rule: Login Attempt**

- **Event:** trackAction call with context.login.attempt = true
- **Actions:**
  - Set event2 (Login Attempt)
  - If context.login.status = success: Set event3 (Login Success), eVar2 = user ID, eVar4 = account name
  - If context.login.status = failure: Set event4 (Login Failure)

4. **Rule: Button / Link Clicked**

- **Event:** trackAction call with context.button.clicked present
- **Actions:**
  - Set eVar2 if user is logged in
  - Set prop2 = context.button.clicked
  - Set event5 (Feature Interaction)

5.2 Data Elements

- **Data Elements:**
  - screenName -> context.screen.name
  - buttonClicked -> context.button.clicked
  - userID -> context.user.id_hashed
  - accountName -> context.user.account_name
  - overdueCount -> context.invoice.overdue_count
  - overdueAmount -> context.invoice.overdue_amount
  - upcomingPayouts -> context.invoice.upcoming_payouts
  - instantPayout -> context.invoice.instant_payout

5.3 Extensions and Configuration

- **Adobe Analytics Extension**:
  - Configure report suite, global variables (e.g., app name), and server calls.
  - Map context data to Analytics variables in the extension's variable mapping panel.
- **Edge Network Extension (for AEP)**:
  - Setup XDM schema and datastream in Launch.
  - Map data elements to XDM fields.

- **Mobile Core Extension**:
  - Ensure lifecycle metrics are enabled.
  - Manage privacy status if needed.

6. Mobile SDK Implementation Details

6.1 Initialization and Launch Configuration

- Integrate the AEP Mobile SDK with Launch using the Configuration ID.
- Ensure the SDK is initialized at app start-up (e.g., in onCreate() method for Android or application(_:didFinishLaunchingWithOptions:) for iOS).

6.2 trackState() and trackAction() Usage

- **trackState(screenName, contextData)**:
  - For login screen: trackState("login_screen", { "context.screen.name": "login_screen" })
  - For dashboard: trackState("dashboard_screen", { "context.screen.name": "dashboard_screen", "context.invoice.overdue_count": "4", ... })
- **trackAction(actionName, contextData)**:
  - For Login Attempt: trackAction("login_attempt", { "context.login.attempt": "true", "context.login.status": "success" })
  - For Button Click: trackAction("button_click", { "context.button.clicked": "create_new" })

6.3 Offline Data and Lifecycle Metrics

- Enable offline tracking if the device is offline. Data will queue and send when connectivity returns.
- Lifecycle metrics automatically collected (launches, crash detection) can be extended if needed.

7. Testing and Validation

- Use Adobe Assurance (Griffon) for real-time validation.
- Validate that the correct variables (eVars, props) and events are set for each interaction.
- Confirm data appears correctly in the Adobe Analytics report suite and in AEP sandbox.
- Test multiple user journeys: failed login attempts, successful logins, multiple navigations to the dashboard, button clicks, etc.

8. Appendix

- **Reporting Dimensions (Example):**
  - Screen Name (eVar1) to see top visited screens.
  - Button Clicked (prop2) to see which CTA is most popular.

- **Reporting Metrics (Example):**
  - event3 (Login Success) to measure conversion rates from the login screen.
  - event5 (Feature Interaction) to measure engagement with dashboard features.

End of SDR

This Solution Design Reference provides a comprehensive blueprint for implementing Adobe Analytics on the Green Pay mobile app screens using the Adobe Launch and Mobile SDK. It includes details on what data to capture, how to capture it, and how to map it into Adobe Analytics and AEP for further analysis and activation.