**BMW Security App (SD3 App)  
Subscription + Payment Management**Technical Specification

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Details** |
| 0.1 | 14h June 2021 | Tommy Ahearne | First Draft |
| **0.2** | **23rd June 2021** | **Nikki Barnett** | **Add subscriptionId Remove fee** |

|  |
| --- |
| **Document Purpose** |
| This document describes the requirements and technical specification to develop and deploy Subscription and Payment management for the BMW Security Smartphone App.  App UX, APIs, CRM and Payment Gateway are all considered.  The solution specified will also form the framework for future SD3 App adoptions i.e. Tracker UK, LJ Italia. |
|  |
| **Overview** |
| The chosen solution for subscription management is Salesforce (SF). SF has extensive capability for customer, product and subscription management and has the advantage of integrations with solutions for Payment, Tax calculation, Campaign Management and Oracle ERP.  The BMW App should be unaware of any SF integration, and all API requests will use passthrough endpoints within the **SDCTC** API, abstracting SF from the mobile app.  The SF Apex API framework will be utilised to create API endpoints for **SDCTC** to consume.  **SDCTC** will be required to authenticate with the SF Apex API using ENV VAR stored credentials, and to send the customer’s SF Account ID with all requests, which can be will be stored within the CTC Account SolomonId field.  *To ease pressure on the* ***SDCTC*** *team, and to expedite the availability of the Passthrough API for the Application team to integrate, Tracker UK will develop and host a version of the Passthrough API. These few Subscription endpoints can be merged into the* ***SDCTC*** *API when capacity is available. We can discuss the format of the API, and will follow the lead of* ***SDCTC.*** |
|  |
| **Payment and Card Management** |
| A third-party “payment gateway aggregator” named **Asperato.com** has been chosen to make subscription plan payments. Asperatointegrates directly with SF, creating a smooth user and system journey.  All payments and card management requests are completed via dedicated and secure Asperato web pages, which can be simply displayed within the BMW App as a web view.  This process simplifies payment management, and delegates sensitive financial detail management to Asperato.  Once payment is successfully completed via a payment webpage within the app, Asperato informs SF, allowing the Subscription to be immediately updated/renewed. The BMW App will present the updated subscription to consumer. |

**API Documentation**

**NOTE: SDCTC** will add the following into the query parameters of all requests:

* {sfid} = the CTC Account SolomonId
* {lang} = i18n locale code for language from CTC Account

|  |  |
| --- | --- |
| **Name** | **Get Subscription Plans** |
| **Description** | Returns a list of available subscription plans for a specific VIN, allowing the app to display as a selection to the consumer. SF will filter the available plans based on account type (BMW, Tracker UK etc) and the specific hardware installed in the vehicle.  *Details of the specific plans will be made available by the business unit.* |
| **Request Verb + Path** | **GET** /subscription/plans |
| **Request Query Parameters** | {vin} = VIN of vehicle |
| **Request Headers** | Standard SDCTC Headers only |
|  |  |
| **Response Body** | **HTTP Success: 200**  [{  "planId": "xxx",  "name": "Example Subscription Name",  "description": "Example Subscription Description",  "termMonths": 36,  "currencySymbol" "£",  "totalPrice": 240.0,  "tax": 40.0  }]  **HTTP Error: 500, 404, 401, 400**  {  "errorCode": "", // **Divya + Ash to Confirm style**  } |
| **Response Properties** | |  |  |  | | --- | --- | --- | | Name | Type | Description | | planId | String | SF identifier for plan | | name | String | Short localised name of plan. For Titles | | description | String | Long localised description of plan | | termMonths | Integer | Number of months for plan | | currencySymbol | String | Symbol of currency to display | | totalPrice | Decimal | Plan price including taxes | | tax | Decimal | Calculated tax for plan | |
| **Application Mocks** | (to be provided) |

|  |  |
| --- | --- |
| **Name** | **Create Plan Payment Url** |
| **Description** | Returns a new customised payment url, which will allow the customer to pay for the plan they have selected. Url can be hosted in WebView/iFrame.   * The success or failure of a payment is not known to the app/webportal but will be displayed to the customer via the payment web page. * Prior to displaying confirmation of payment to the customer, Asperato will call SF to update the subscription status. * The app/portal should request the updated Subscription immediately after the payment webpage /iframe is closed. This will allow the app to keep in sync with the subscription in Salesforce. * If the payment failed, the app will be unaware, status on subscription will simply remain the same. |
| **Request Verb + Path** | **POST** /subscription/plans/paymenturl |
| **Request Query Parameters** |  |
| **Request Headers** | Standard SDCTC Headers only |
| **Request Body** | {  "planId": "xxx",  "subscriptionId": "xxxxxxxxxxxxxxxxxx",  "vin": "xxxxxxxxxxxxxxx",  } |
|  |  |
| **Response Body** | **HTTP Success: 200**  {  "paymentUrl": "https://asperato.com/XXXXXXXX",  }  **HTTP Error: 500, 409, 404, 401, 400**  {  "errorCode": "", // **Divya + Ash to Confirm style**  } |
| **Response Properties** | |  |  |  | | --- | --- | --- | | Name | Type | Description | | paymentUrl | String | Url for customised payment page for transaction. This should be launched in an app webview or iFrame in webportal | |
| **Application Mocks** | (to be provided) |

|  |  |
| --- | --- |
| **Name** | **Get Subscriptions** |
| **Description** | Returns a list of all subscriptions within a customer’s account (for all vehicles). Each subscription will be mapped to an asset via the VIN property.   * **Plans:** Each subscription will have a “Current Plan” and a “Renewal Plan” (upcoming). * The “current plan” can have a status of [Active, Expired, RecoveryMode\*]. * The “Renewal Plan” will be null, unless the customer has renewed their plan before the current plan has expired. The renewal plan can be considered the “Upcoming Plan” and should be displayed in the app with the detail of the plan they have chosen. * If a “renewal plan” exists at the expiry date of the current plan, the current plan will be replaced with the renewal plan. * The customer can choose to renew a plan at anytime during the term of the current plan, or after the current plan has expired. Payment is taken immediately for any renewal. * **Cancellation**: If the customer choses to end their subscription, they must wait for the term of the current subscription to expire and not renew. Customers can request their account to be closed immediately via support option (*contact details should be added in the app)*, which will put their account through a background deactivation process, not documented here. * **Refunds** are not offered via the app. Refunds are at the discretion of the business and can be given directly through SF and Asperato by customer service * **Auto-Renewal/CPA** will not be considered for the initial release of the product. *The expectation is that payment for over 70% of initial subscriptions will be taken at the BMW dealership. In these cases, auto renewal is not possible and could only be setup as part of the first renewal within the app. This will be reviewed at a later stage.* * **Features:** A CSV ofavailable features will be present with the subscription plan. The app and web portal will display UI features based on an existing key in the CSV * **App Display:**    + The app should display on the dashboard a notice of upcoming expiry (if a renewal plan in not in place), and a notice when the subscription has expired. With a CTA to take the customer to the plans to renewal.   + When expired, the functionality for the expired vehicle should be completely disabled, only allowing renewal   + When in recovery mode, the app should not allow the user to interact with any of the features, and should display that the app has had the location features removed to protect the customer. |
| **Request Verb + Path** | **GET** /subscription |
| **Request Query Parameters** |  |
| **Request Headers** | Standard SDCTC Headers only |
|  |  |
| **Response Body** | **HTTP Success: 200**  [  {  "subscriptionId": "xxxxxxxxxxxxxxxxxx",  "vin": "xxxxxxxxxxxxxxx",  "currentPlan": {  "planId": "xxx",  "name": "Example Subscription Name",  "description": "Example Description",  "termMonths": 36,  "startDate": "2021-01-01",  "expiryDate": "2024-01-01",  "statusId": 1, 1=Active, 2=Expired, 3=RecoveryMode,  "features": "PLACES,BATTHEALTH,TRIPS", **TBD**  },  "renewalPlan": {  "planId": "xxx",  "name": "Example Subscription Name",  "description": "Example Subscription Description",  "termMonths": 36,  "startDate": "2021-01-01",  "expiryDate": "2024-01-01"  }  }  ]  **HTTP Error: 500, 404, 401, 400**  {  "errorCode": "", // **Divya + Ash to Confirm style**  } |
| **Response Properties** | |  |  |  | | --- | --- | --- | | Name | Type | Description | | vin | String | The VIN to which the subscription belongs | | currentPlan | obj | The existing in use. The plan could be expired or active. ***The plan will not be null*** | | renewalPlan | obj | The queued plan to replace the currentPlan once expired. Could be null if no queued plan. | | planId | String | SF identifier for plan | | name | String | Short localised name of plan. For Titles | | description | String | Long localised description of plan | | termMonths | Integer | Number of months for plan | | startDate | ISO Date | Date of which the plan started | | expiryDate | ISO Date | Date after the last date of current plan. | | statusId | integer | Status of the current plan. Active, Expired or RecoveryMode | | features | String (cvs) | The features list (csv) inside the current plan defines the features available to the UI. These features are subscription driven. List TBD | |
| **Application Mocks** | (to be provided) |
|  |  |