

# gm\_ewarr\_sale\_t2 – Comprehensive Specification

Unified reference for business stakeholders, architects, and delivery teams

**Source System:** gm\_ewarr\_sale\_t2.fmb

## 1. Business Summary

### Purpose

The **GM\_EWARR\_SALE** Oracle Forms application facilitates the management and processing of extended warranty sales for vehicles. It enables users to capture detailed warranty contract information, validate vehicle and customer data, calculate premiums, handle payment modes including online and EMI options, and manage associated documents and add-on packages. The form integrates loyalty program features such as OTP verification and points redemption to enhance customer engagement.

### Key Stakeholders

- **Sales Team:** Responsible for entering and managing extended warranty contracts and customer information.
- **Service Advisors / DSEs:** Use the application to validate vehicle details, assist customers with warranty options, and manage service-related data.
- **Customers:** Beneficiaries of extended warranty services and loyalty programs, interacting indirectly through sales and service representatives.
- **Finance Department:** Oversees payment processing, premium calculations, and invoicing related to warranty sales.
- **IT Support and Developers:** Maintain and enhance the Oracle Forms application, manage WebUtil integrations, and ensure system reliability.
- **Compliance and Audit Teams:** Ensure data integrity, proper documentation upload, and adherence to business rules and regulatory requirements.

### Business Value

This application streamlines the extended warranty sales process, ensuring accurate data capture and validation to reduce errors and improve customer satisfaction. It supports

multiple payment modes, including online payments with OTP verification, enhancing payment security and convenience. The integration of document management ensures compliance and easy retrieval of warranty-related documents. Additionally, the loyalty program integration fosters customer retention by allowing points redemption and rewards. Overall, the system reduces manual errors, accelerates warranty processing, and enhances transparency and traceability.

## Success Metrics

- **Accuracy of Warranty Data:** Reduction in data entry errors and validation failures.
- **Customer Satisfaction:** Positive feedback on ease of warranty purchase and document handling.
- **Transaction Throughput:** Number of warranty contracts processed per day with minimal rework.
- **Payment Completion Rate:** Successful online payment transactions and OTP validations.
- **Document Compliance:** Percentage of warranty cases with required minimum document uploads (minimum 4).
- **Loyalty Program Engagement:** Number of customers redeeming points and validating OTPs successfully.
- **System Stability:** Minimal runtime errors and prompt error handling via form-level triggers.

## High-Level Process Flow

### 1. Contract Entry

User inputs the extended warranty contract number and payment mode. The system validates the contract and fetches related vehicle and customer details.

### 2. Vehicle & Customer Validation

VIN and warranty options are validated with business rules. Loyalty card details and customer information are verified. Errors and warnings are displayed as needed.

### 3. Premium Calculation

Premium amounts are calculated automatically, considering add-on packages and applicable taxes. Users

### 4. OTP Verification & Loyalty Redemption

For loyalty program participants, OTP is sent and confirmed. Redeemable points are calculated and applied to

can trigger premium recalculation via dedicated buttons.

premium amounts upon successful OTP validation.

## **5. Document Management**

Users upload necessary warranty documents, with validations ensuring minimum document count and file size limits. Document download and remarks entry are supported.

## **6. Payment Processing**

Payment mode selection includes cash and online options. Online payments trigger notifications to customers with payment links. EMI payment mode includes special handling and messaging.

## **7. Data Commit & Reporting**

On commit, data is validated and saved. Reports such as warranty certificates and add-on package reports can be generated and printed.

## **8. Error Handling & Logging**

Form-level triggers capture errors and messages, displaying user-friendly alerts. Application logging is optionally enabled for performance and audit tracking.

## 2. Comprehensive Functional Requirements

### Functional Requirements for GM\_EWARR\_SALE\_T2

#### 1. Core Workflows

Workflow	Description	Key Steps	Triggers / Program Units Involved
Extended Warranty Contract Search and Retrieval	Allows users to search and retrieve extended warranty contract details using contract number.	<ol style="list-style-type: none"><li>User enters Contract No. in <code>NB_EWARR_NUM</code> field.</li><li>System validates input and fetches contract details from database.</li><li>Populate warranty, vehicle, customer, and payment details on form.</li><li>Display error messages for invalid or not found contract numbers.</li></ol>	<ul style="list-style-type: none"><li><code>WHEN-VALIDATE-ITEM</code> on <code>CB_VT_EWARR_SALE.NB_EWARR_NUM</code></li><li><code>PROC_CHECK_NULL</code> procedure</li><li><code>ON-MESSAGE</code> trigger for error handling</li></ul>

Workflow	Description	Key Steps	Triggers / Program Units Involved
Premium Calculation	<p>Calculate extended warranty premium based on vehicle details, warranty options, and addons.</p>	<ol style="list-style-type: none"> <li>1. User selects warranty options and payment mode.</li> <li>2. User clicks <b>Calculate Premium</b> button.</li> <li>3. System calculates premium including applicable taxes.</li> <li>4. Display calculated premium and EMI text if applicable.</li> </ol>	<ul style="list-style-type: none"> <li>• <code>WHEN-BUTTON-PRESSED</code> on <code>PB_CAL_PRE</code></li> <li>• <code>PKG_EXTE_WAR.EW_CALC_PREM_GST</code> called internally</li> </ul>

Workflow	Description	Key Steps	Triggers / Program Units Involved
Payment Mode Selection and Validation	Manage payment mode selection and related validations including EMI and online payment options.	<ol style="list-style-type: none"> <li>1. User selects payment mode from <code>PAY_MODE</code> list.</li> <li>2. System validates payment mode and updates related fields.</li> <li>3. Enable or disable EMI information based on payment mode.</li> <li>4. Display informational messages or errors if invalid selection.</li> </ol>	<ul style="list-style-type: none"> <li>• <code>WHEN-VALIDATE-ITEM</code> and <code>WHEN-LIST-CHANGED</code> on <code>PAY_MODE</code></li> <li>• <code>ON-MESSAGE</code> trigger for error handling</li> </ul>

Workflow	Description	Key Steps	Triggers / Program Units Involved
OTP (One-Time Password) Generation and Validation	Send OTP to customer and validate entered OTP for loyalty point redemption.	<ol style="list-style-type: none"> <li>1. User clicks <code>Send/Resend OTP</code> button.</li> <li>2. System generates OTP and sends SMS to registered mobile number.</li> <li>3. User enters received OTP and clicks <code>Confirm OTP</code>.</li> <li>4. System validates OTP and updates loyalty redemption status.</li> <li>5. Display success or error messages accordingly.</li> </ol>	<ul style="list-style-type: none"> <li>• <code>WHEN-BUTTON-PRESSED</code> on <code>PB OTP_SEND</code> and <code>PB CNF OTP</code></li> <li>• <code>PKG_LOYALTY.SP_SEND_SMS</code> and OTP validation queries</li> <li>• <code>GENERAL.DISP_TEXT</code> for messaging</li> </ul>

Workflow	Description	Key Steps	Triggers / Program Units Involved
Document Upload and Management	Allow users to upload, save, and download vehicle and warranty related documents.	<ol style="list-style-type: none"> <li>1. User navigates to document upload block.</li> <li>2. User selects files and clicks <code>Upload</code> button.</li> <li>3. System validates file size, name, and uploads file to server.</li> <li>4. User can save uploaded documents metadata and download files.</li> <li>5. System enforces minimum 4 documents upload before saving.</li> </ol>	<ul style="list-style-type: none"> <li>• <code>WHEN-BUTTON-PRESSED</code> on <code>UPLOAD</code>, <code>SAVE</code>, and <code>DOWNLOAD</code> buttons in <code>B_EW_DOCS</code></li> <li>• <code>UPLOAD_FILE</code> and <code>DOWNLOAD_FILE</code> procedures</li> <li>• <code>WHEN-NEW-RECORD-INSTANCE</code> and <code>PRE-INSERT</code> triggers on <code>B_EW_DOCS</code></li> <li>• <code>GENERAL.DISP_TEXT</code> for user notifications</li> </ul>

Workflow	Description	Key Steps	Triggers / Program Units Involved
Addon Package Selection and Pricing	<p>Manage customer convenience package (CCP) selection and calculate related charges.</p>	<ol style="list-style-type: none"> <li>1. User selects addon packages via checkbox in addon block.</li> <li>2. System calculates addon basic amount, taxes, discounts, and total.</li> <li>3. User can navigate back or save addon selections.</li> <li>4. System validates addon selections and displays messages on errors.</li> </ol>	<ul style="list-style-type: none"> <li>• WHEN-CHECKBOX-CHANGED on ADDON_YN</li> <li>• WHEN-BUTTON-PRESSED on ADDON_BACK</li> <li>• CALL_REPORT_ADDON procedure for reports</li> <li>• PKG_ADDON_SALE.CALC_PREM for premium calculation</li> </ul>

Workflow	Description	Key Steps	Triggers / Program Units Involved
Form Initialization and Error Handling	Initialize form state and handle errors during form usage.	<ol style="list-style-type: none"> <li>1. On form open, initialize global variables and logging timer.</li> <li>2. Handle errors such as invalid data, concurrency conflicts, and validation failures.</li> <li>3. Display user-friendly messages using alerts and display text.</li> <li>4. Provide clear form functionality resetting the form to initial state.</li> </ol>	<ul style="list-style-type: none"> <li>• <code>PRE-FORM</code> and <code>ON-ERROR</code> form-level triggers</li> <li>• <code>KEY-CLRFRM</code> trigger to clear form</li> <li>• <code>GENERAL.DISP_TEXT</code> and <code>ALERT</code> objects</li> </ul>

Workflow	Description	Key Steps	Triggers / Program Units Involved
Report Generation	Generate extended warranty and addon package certificates and reports.	<ol style="list-style-type: none"> <li>1. User triggers report generation after contract save or on demand.</li> <li>2. System calls report procedures to generate PDF or print outputs.</li> <li>3. Display confirmation alerts before report generation.</li> </ol>	<ul style="list-style-type: none"> <li>• <code>CALL_REPORT_EW</code> and <code>CALL_REPORT_ADDON</code> procedures</li> <li>• <code>AL_REPORT</code> and <code>AL_REPORT1</code> alerts</li> <li>• Report file: <code>vr_ew_certificate.rdf</code></li> </ul>

## 2. Detailed Functional Coverage

### Extended Warranty Contract Management

- Search contracts by contract number.
- Display vehicle, customer, and warranty details.
- Validate contract data integrity and concurrency.
- Support warranty type changes and effective dates.
- Maintain audit fields: created/modified by and dates.

### Payment Processing and Validation

- Support multiple payment modes: cash, online, EMI.
- Validate payment mode selection and related fields.
- Display EMI text notices for online EMI payments.
- Calculate total premiums including GST components.
- Handle loyalty points redemption and award calculations.

## **OTP Handling for Loyalty Program**

- Generate and send OTP to registered mobile.
- Validate OTP entered by user against database.
- Update loyalty redemption status upon successful OTP confirmation.
- Display error messages for invalid or expired OTP.
- Integrate with loyalty point calculation procedures.

## **Document Upload and Download Management**

- Upload vehicle and warranty related documents.
- Enforce minimum document upload count (min 4).
- Validate file size and name constraints.
- Save document metadata with audit information.
- Support document download and back navigation.
- Handle errors during upload/download gracefully.

## **Addon Package Selection and Pricing**

- Display list of available convenience packages.
- Allow user to select/deselect addon packages.
- Calculate addon pricing including taxes and discounts.
- Provide back navigation and save functionality.
- Validate addon selections against business rules.

## **Form Interaction and Navigation**

- Manage navigation between blocks and items.
- Enable/disable fields dynamically based on context.
- Clear form and reset states on user command.
- Handle keyboard events like list validation and exit.
- Show alerts and messages contextually.

## **Reporting and Certificate Generation**

- Provide extended warranty and addon package reports.
- Support user prompts before report

## **Error Handling and Logging**

- Capture and display database and application errors.
- Handle validation errors with user-friendly messages.

generation.

- Integrate with Oracle Reports via procedures.
- Allow printing and viewing of generated reports.

- Manage concurrency conflicts with rollback and alerts.

- Log application activity conditionally based on global flags.
- Use form-level triggers to centralize error processing.

### 3. Business Rules Matrix

Rule ID	Business Rule	Applies To	Validation / Action	Trigger / Procedure
BR-001	Contract number must be valid and exist in the system.	NB_EWARR_NUM field	Validate input and fetch contract details; error if not found.	WHEN-VALIDATE-ITEM on NB_EWARR_NUM
BR-002	Payment mode must be selected; EMI mode disables certificate generation.	PAY_MODE field	Enable/disable EMI text and certificate generation accordingly.	WHEN-VALIDATE-ITEM and WHEN-LIST-CHANGED on PAY_MODE
BR-003	Minimum four documents must be uploaded before saving.	Document upload block (B_EW_DOCS)	Count uploaded docs; prevent save if less than 4.	WHEN-BUTTON-PRESSED on SAVE button of B_EW_DOCS
BR-004	OTP must be validated before redeeming loyalty points.	OTP confirmation process	Check OTP against log; show error for invalid or expired OTP.	WHEN-BUTTON-PRESSED on PB_CNF OTP
BR-005	Addon package selections must be validated and pricing recalculated.	Addon checkbox and pricing fields	Recalculate totals on selection change; disallow invalid combos.	WHEN-CHECKBOX-CHANGED on ADDON_YN

Rule ID	Business Rule	Applies To	Validation / Action	Trigger / Procedure
BR-006	Vehicle VIN must be validated for warranty eligibility.	VIN field in B_VT_EWARR_SALE	Verify VIN existence and eligibility; prevent invalid entries.	WHEN-VALIDATE-ITEM on VIN
BR-007	Redeemed points cannot exceed available loyalty balance.	LOYL_REDEEM PTS field	Validate points against balance; error if exceeded.	WHEN-VALIDATE-ITEM on LOYL_REDEEM PTS
BR-008	Form must be cleared and reset on clear command.	Form clear action	Reset all fields, enable/disable controls per default state.	KEY-CLRFRM trigger on form
BR-009	System must handle concurrency conflicts gracefully.	All database updates	Detect update conflicts; rollback and notify user.	ON-ERROR trigger on form
BR-010	Extended warranty premium calculation must include GST components.	Premium calculation process	Calculate base premium plus CGST, SGST, IGST taxes.	PB_CAL_PRE WHEN-BUTTON-PRESSED trigger and PKG_EXTE_WAR.EW_CALC_PREM_GST

### 3. Technical Specifications

#### Solution Architecture

##### Overview

The **GM\_EWARR\_SALE** Oracle Forms application is designed to manage extended warranty sales, including contract details, payment modes, document uploads, and add-on packages. It integrates with multiple database tables such as *EW\_DOCS* and uses WebUtil for enhanced client-side functionalities.

##### Key Components

- **Data Blocks:** 5 blocks including main sales data, warranty details, documents, add-ons, and WebUtil integration.
- **Triggers:** 50 triggers managing form-level events, validation, button actions, and custom logic.
- **Program Units:** 9 PL/SQL procedures and functions handling business logic like premium calculation, file upload/download, and report generation.
- **LOVs:** 2 List of Values for document selection and employee lookup.
- **Alerts:** 4 alerts for error handling, confirmations, and warnings.
- **Attached Libraries:** WEBUTIL for client integration.
- **Reports:** One report for extended warranty certificates.

##### Integration & Extensibility

The form leverages WebUtil library for client-side operations such as file handling and system info retrieval. Program units encapsulate reusable business logic, allowing easy maintenance and extension.

## API Surface

Program Unit	Type	Description
PROC_CHECK_NULL	Procedure	Validates mandatory fields for null values.
UPLOAD_FILE	Procedure	Handles file upload to server directories.
CALL_REPORT_EW	Procedure	Generates extended warranty reports.
EXT_WAR_MSG	Function	Returns messages related to extended warranty processing.
GEN_EINV_QR_CODE	Procedure	Generates QR codes for electronic invoices.
GEN_CSV_PWC	Procedure	Generates CSV files for premium with calculation.
DOWNLOAD_FILE	Procedure	Handles file download from server to client.
CALL_REPORT_ADDON	Procedure	Generates reports for add-on packages.
GEN_EINV_QR_CODE_CCP	Procedure	Generates QR codes for CCP electronic invoices.

## Oracle Forms Elements

### Form Level Triggers

Trigger Name	Event	Description	Code Length (Lines)
ON-ERROR	ON-ERROR	Handles form errors with customized messages and rollback logic.	1730
ON-MESSAGE	ON-MESSAGE	Displays specific error messages for error codes 40209 and 40207.	343
KEY-EXIT	KEY-EXIT	Cleans up session info and exits the form.	155
KEY-LISTVAL	KEY-LISTVAL	Handles LOV display and validation on list value keys.	407
KEY-COMMIT	KEY-COMMIT	Manages commit logic with report generation and validations.	3332
KEY-CLRFRM	KEY-CLRFRM	Clears the form and resets item properties.	2697
WHEN-TIMER-EXPIRED	WHEN-TIMER-EXPIRED	Timer event handler (minimal code).	27
PRE-FORM	PRE-FORM	Initializes application info and logging.	267

## Data Blocks

Block Name	Data Source	Items Count	Description
CB_VT_EWARR_SALE	N/A	11	Controls contract number, payment mode, premium calculation, OTP, document upload and add-on package buttons.
B_VT_EWARR_SALE	N/A	132	Main extended warranty details including vehicle info, customer details, premium, loyalty points, and contract metadata.
B_EW_DOCS	EW_DOCS	25	Manages document metadata and upload/download functionality for warranty contracts.
B_ADDON	N/A	20	Handles customer convenience package add-on details and pricing.
WEBUTIL	N/A	9	Integrates WebUtil bean areas for client info, file handling, session, OLE, API, and browser functions.

## LOVs (List of Values)

LOV Name	Record Group	Title	Return Item(s)
LOV_DOCS	RG_DOCS	Select Document From List.....!!	B_EW_DOCS.REMARKS (Documents)
LV_EMP	RG_EMP	Service Advisor List	B_VT_EWARR_SALE.NB_EMP_NAME, B_VT_EWARR_SALE.EXTE_EMP_CD

## Program Units

- **PROC\_CHECK\_NULL** - Validates mandatory fields.
- **UPLOAD\_FILE** - Handles file upload processing.
- **CALL\_REPORT\_EW** - Generates extended warranty reports.
- **EXT\_WAR\_MSG** - Returns extended warranty related messages.
- **GEN\_EINV\_QR\_CODE** - Generates QR codes for e-invoices.
- **GEN\_CSV\_PWC** - Generates CSV files for premium with calculation.

- **DOWNLOAD\_FILE** - Handles file download operations.
- **CALL\_REPORT\_ADDON** - Generates reports for add-on packages.
- **GEN\_EINV\_QR\_CODE\_CCP** - Generates QR codes for CCP invoices.

## Alerts

Alert Name	Title	Message	Alert Style
WEBUTIL_ERROR			STOP
AL_REPORT	Extended Warranty	Do you want to?	Note
AL_REPORT1	Certificate Selection	Please select option	Note
AL_SEGMENT		You have Selected Wrong Segment and Part Combination. Do you want to proceed ?	Caution

## Attached Libraries

- **WEBUTIL** - Provides client integration features like file handling, OLE, and session management.

## Canvases

Canvas Name	Width (px)	Height (px)	Description
CN_ERR	N/A	N/A	Error display canvas
CN_DOCS	740	193	Document management canvas
CN_ADDON	734	286	Add-on package management canvas
WEBUTIL_CANVAS	283	214	WebUtil integration canvas for client features

## Record Groups

Record Group Name	Type	Query Enabled
RG_DOCS	N/A	No
RG_OPTION_CD	Query	No
RG_EWR_TYPE	Query	No
RG_EMP	Query	No

## Reports

Report Name	File Path
REPORT_CALL	D:\reports\samples\demo\vr_ew_certificate.rdf

## Visual Attributes

Visual Attribute Name	Font Name	Font Size	Foreground Color	Background Color
VA_DEFAULT	N/A	N/A	N/A	N/A
VA_MANDATORY	N/A	N/A	N/A	N/A

## Windows

- **WEBUTIL\_HIDDEN\_WINDOW** - A hidden window used by WebUtil for client integration, sized 282x214 pixels.

## Validation, Security & Operations

- **Validation** Extensive item-level and block-level validation triggers ensure data integrity, including mandatory checks, format validations, and business rule enforcement.
- **Security** Error handling triggers prevent unauthorized or invalid data entry. OTP validation and loyalty program checks secure transactional authenticity.
- **Operations** File upload/download management with size and path validations, integration with external procedures for report generation, and premium calculations.
- **User Experience** Use of alerts and messages for user guidance, dynamic enabling/disabling of fields based on context (e.g., payment mode), and seamless

navigation between data blocks.

- **Logging & Monitoring** Application info and module logging initialized in PRE-FORM triggers for audit and performance monitoring.

## 4. Business Logic & Flows

### Business Flows & Sequence

#### Form Initialization Flow

1. Form-level triggers execute (0 triggers)
2. Data blocks initialized (0 blocks)
3. LOVs loaded (0 LOVs)
4. Program units available (0 units)

#### User Interaction Flow

1. User interacts with form items
2. Block-level triggers validate input
3. LOVs provide selection options
4. Program units execute business logic
5. Alerts notify user of issues (0 alerts)

#### Data Processing Flow

1. Record groups fetch data (0 groups)
2. Data blocks query database
3. Program units process calculations
4. Reports generated (0 reports)

## Procedures

Procedure Name	Details
<b>WebUtil_Core.ShowBeans</b>	Stored procedure or function
<b>SHOWBEANS</b>	Stored procedure or function
<b>WEBUTIL_CORE.CustomEventHandler</b>	Stored procedure or function
<b>CUSTOMEVENTHANDLER</b>	Stored procedure or function
<b>VARCHAR2</b>	Stored procedure or function
<b>in</b>	Stored procedure or function
<b>DISP_TEXT</b>	Stored procedure or function
<b>TO_CHAR</b>	Stored procedure or function
<b>set_client_info</b>	Stored procedure or function
<b>SET_MODULE</b>	Stored procedure or function
<b>GENERAL.DISP_TEXT</b>	Stored procedure or function
<b>REPLACE</b>	Stored procedure or function
<b>INSTR</b>	Stored procedure or function
<b>WEBUTIL_FILE.FILE_SIZE</b>	Stored procedure or function
<b>CLIENT_GET_FILE_NAME</b>	Stored procedure or function
<b>FILE_SIZE</b>	Stored procedure or function
<b>TO_NUMBER</b>	Stored procedure or function
<b>NVL</b>	Stored procedure or function
<b>UPPER</b>	Stored procedure or function
<b>SUBSTR</b>	Stored procedure or function
<b>LOWER</b>	Stored procedure or function

Procedure Name	Details
<b>DOWNLOAD_FILE</b>	Stored procedure or function
<b>GO_ITEM</b>	Stored procedure or function
<b>GO_BLOCK</b>	Stored procedure or function
<b>UPLOAD_FILE</b>	Stored procedure or function
<b>clear_form</b>	Stored procedure or function
<b>SET_ITEM_PROPERTY</b>	Stored procedure or function
<b>IN</b>	Stored procedure or function
<b>format</b>	Stored procedure or function
<b>RAISE_ERROR</b>	Stored procedure or function
<b>SET_CLIENT_INFO</b>	Stored procedure or function
<b>EXIT_FORM</b>	Stored procedure or function
<b>go_block</b>	Stored procedure or function
<b>CLEAR_BLOCK</b>	Stored procedure or function
<b>DECODE</b>	Stored procedure or function
<b>MAX</b>	Stored procedure or function
<b>PKG_LOYALTY.SP_GET_VIN_LOYALTY_DTL</b>	Stored procedure or function
<b>SP_GET_EMPLOYEE_NAME</b>	Stored procedure or function
<b>SP_GET_VIN_LOYALTY_DTL</b>	Stored procedure or function
<b>GENERAL.CHECK_NULL</b>	Stored procedure or function
<b>CHECK_NULL</b>	Stored procedure or function
<b>MONTHS_BETWEEN</b>	Stored procedure or function
<b>TRUNC</b>	Stored procedure or function

Procedure Name	Details
<b>PKG_EXTE_WAR.EW_CALC_PREM_GST</b>	Stored procedure or function
<b>PKG_LOYALTY.SF_CALCULATE_RUPEES_TO_POINTS</b>	Stored procedure or function
<b>PKG_ADDON_SALE.SP_VIN_VALIDATE</b>	Stored procedure or function
<b>PKG_ADDON_SALE.CALC_PREM</b>	Stored procedure or function
<b>BETWEEN</b>	Stored procedure or function
<b>EW_CALC_PREM_GST</b>	Stored procedure or function
<b>SF_CALCULATE_RUPEES_TO_POINTS</b>	Stored procedure or function
<b>SP_VIN_VALIDATE</b>	Stored procedure or function
<b>CCP</b>	Stored procedure or function
<b>CALC_PREM</b>	Stored procedure or function
<b>PKG_LOYALTY.SF_CALCULATE_POINTS_TO_RUPEES</b>	Stored procedure or function
<b>pkg_loyalty.sp_send_sms</b>	Stored procedure or function
<b>SF_CALCULATE_POINTS_TO_RUPEES</b>	Stored procedure or function
<b>SP_SEND_SMS</b>	Stored procedure or function
<b>NUMBER</b>	Stored procedure or function
<b>CustomEventHandler</b>	Stored procedure or function

## Fields

Field Name	Type
NB_EWARR_NUM	Form field
PAY_MODE	Form field
TOTAL_PREMIUM	Form field
COPY_RIGHT	Form field
STATUS	Form field
EMI_TEXT	Form field
PB_CAL_PRE	Form field
PB OTP SEND	Form field
PB CNF OTP	Form field
PB UPLOAD DOCS	Form field
PB ADDON	Form field
EXTE_POLICY_NO	Form field
EXTE_PREM_SRV_INV_NO	Form field
EXTE_CONTRACT_DATE	Form field
EXTE_MODL_CODE	Form field
EXTE_VARIANT_CD	Form field
EXTE_CHASSIS_NO	Form field
EXTE_ENGINE_NO	Form field
EXTE_REGISTRATION_NO	Form field
VIN	Form field
EXTE_VIN_NO	Form field

Field Name	Type
OEM_VIN_NUMBER	Form field
NB_EXTE_WARR_TYPE_OLD	Form field
NB_EXTE_WARR_TYPE_NEW	Form field
EXTE_WARR_TYPE	Form field
EXTE_SERV_MODL	Form field
NB_EWR_PUR_MILEAGE	Form field
EXTE_CONTRACT_MILEAGE	Form field
EXTE_EMP_CD	Form field
DUMMY_MILEAGE	Form field
EXTE_VALID_DATE	Form field
EXTE_START_DATE	Form field
EXTE_VALID_MILEAGE	Form field
EXTE_PARENT_GROUP	Form field
EXTE DEALER_MAP_CD	Form field
EXTE_LOC_CD	Form field
EXTE_CANCEL_FLAG	Form field
EXTE_COMP_FA	Form field
EXTE_POLICY_DELR	Form field
EXTE_POLICY_FOR	Form field
EXTE_POLICY SOLD_FROM	Form field
EXTE_POLICY_OUTLET	Form field
DLDS_DELR_CODE_DLFN	Form field

Field Name	Type
DLDS_FOR_CODE_DLFN	Form field
DLDS_OUTLET_CODE_DLFN	Form field
EXTE_RETLSALE_DT	Form field
EXTE_CUST_CD	Form field
EXTE_CUST_NAME	Form field
EXTE_CUST_ADD1	Form field
EXTE_CUST_ADD2	Form field
EXTE_CUST_ADD3	Form field
EXTE_CUST_CITY	Form field
CITY_DESC	Form field
EXTE_CUST_EMAIL	Form field
EXTE_CUST_PHONE	Form field
EXTE_BOOKLET_NO	Form field
EXTE_BANK_NAME	Form field
EXTE_CHEQUE_NO	Form field
EXTE_INFAVOUR_OF	Form field
EXTE_CHEQUE_DATE	Form field
EXTE_PREMIUM_RCVD	Form field
EXTE_PREMIUM	Form field
EXTE_PREM_DLR_COMM	Form field
EXTE_PREM_SRV_TAX	Form field
EXTE_PREM_SBC_TAX	Form field

Field Name	Type
EXTE_PREM_KKC_TAX	Form field
EXTE_PREMIUM_CALCULATED	Form field
EXTE_TOT_PREM_SRV_TAX	Form field
EXTE_CUST_PHONE2	Form field
EXTE_CUST_MOBILE	Form field
EXTE_CUST_PIN	Form field
EXTE_CREATED_BY	Form field
EXTE_CREATED_DATE	Form field
MODIFIED_BY	Form field
MODIFIED_DATE	Form field
PRINCIPAL_MAP_CD	Form field
NB_DELIVERY_DATE	Form field
NB_LOCATION_CD	Form field
NB DEALER_MAP_CD	Form field
NB_VEH_SELLING DEALER	Form field
NB_WAR_SELLING DEALER	Form field
NB_MODEL_DESC	Form field
NB_VARIANT_DESC	Form field
NB_OUT_AMT	Form field
NB_COLOUR_CD	Form field
NB_COLOUR_DESC	Form field
EXTE_TOT_PREM_SBC_TAX	Form field

Field Name	Type
EXTE_TOT_PREM_KKC_TAX	Form field
NB_CUST_STATE_CD	Form field
EXTE_CUST_STATE	Form field
CUST_GST_NUM	Form field
PLACE_OF_SUPPLY	Form field
GST_STATE_CD	Form field
SAC_CODE	Form field
GST_TYPE	Form field
NB_GST_TYPE	Form field
CONV_GST_TYPE	Form field
EXTE_PREM_CGST_RATE	Form field
EXTE_PREM_SGST_RATE	Form field
EXTE_PREM_IGST_RATE	Form field

No form-level triggers found.

No validations found.

## 5. Entity Relationship Diagram (ERD)

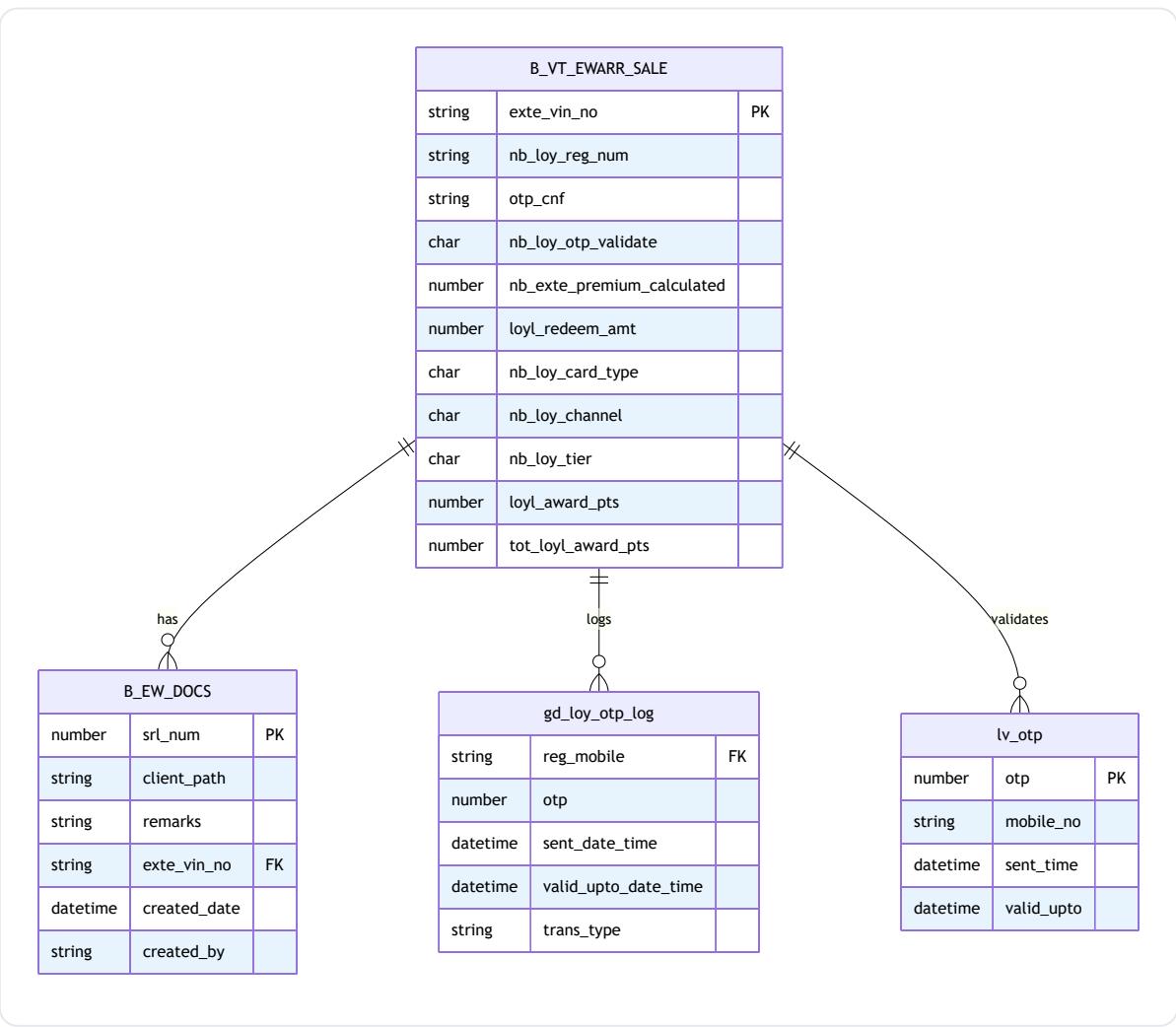
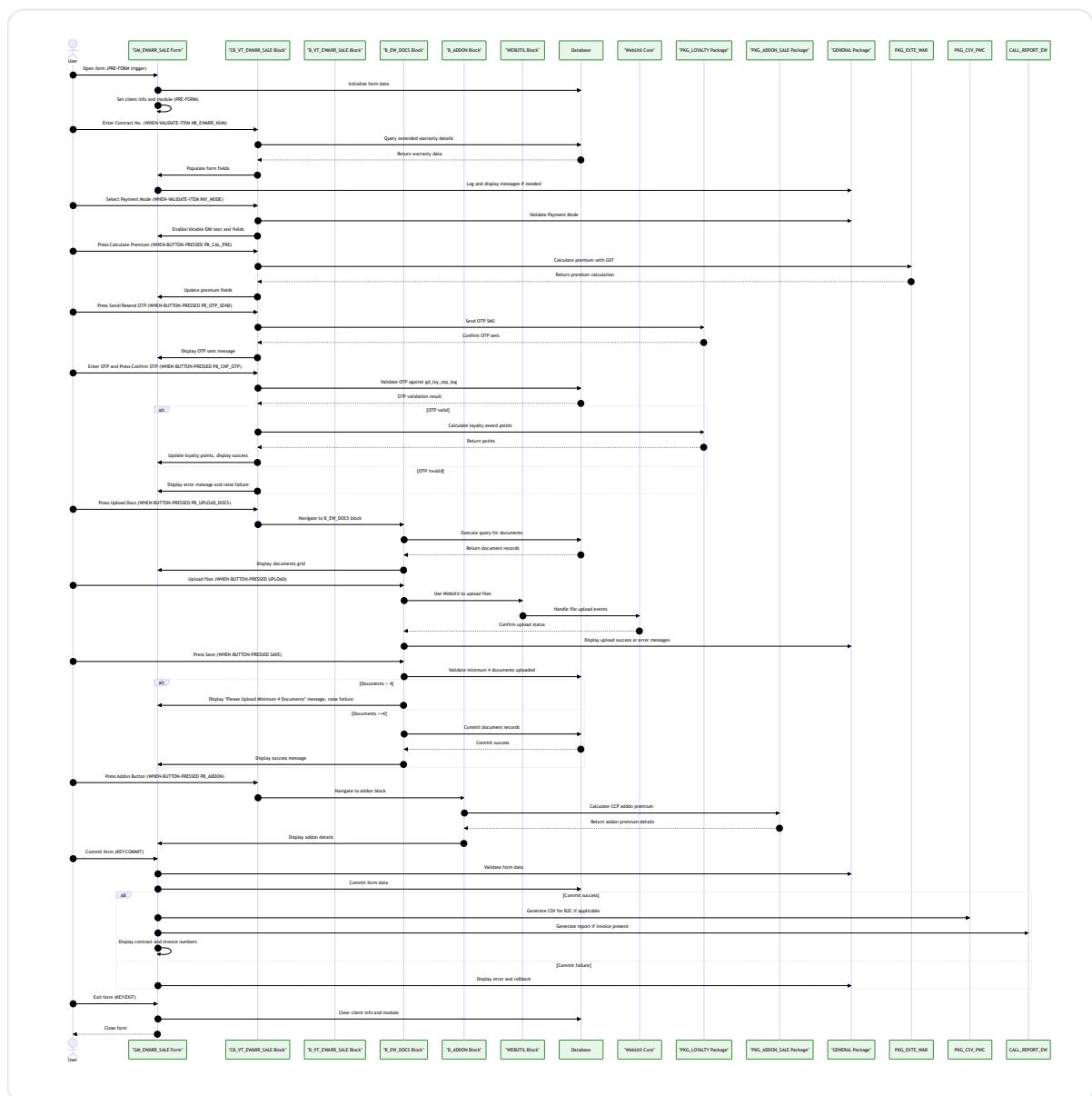


Table	Relationship
B_VT_EWARR_SALE	One-to-Many with B_EW_DOCS (via EXTE_VIN_NO)
B_VT_EWARR_SALE	One-to-Many with gd_loy_otp_log (via nb_loy_reg_num = reg_mobile)
B_VT_EWARR_SALE	One-to-Many with lv_otp (assumed relationship for OTP validation)

### Operational Notes:

- B\_VT\_EWARR\_SALE is the main entity representing warranty sales with VIN as primary key.
- B\_EW\_DOCS stores documents related to a warranty sale and references B\_VT\_EWARR\_SALE by EXTE\_VIN\_NO.
- gd\_loy\_otp\_log holds OTP logs for loyalty verification linked by registered mobile number.
- lv\_otp is assumed to represent OTP details used in validation, linked to B\_VT\_EWARR\_SALE.
- Relationships are one-to-many from B\_VT\_EWARR\_SALE to the other tables.
- OTP confirmation and validation logic is implemented via gd\_loy\_otp\_log and lv\_otp data.
- Document upload and download features are managed in B\_EW\_DOCS with linkage to warranty via VIN.

## 6. Sequence Diagram



**Operational Notes:**

- The form initializes with PRE-FORM trigger setting client and module info.
- User inputs contract number which triggers data fetch and populates warranty details.
- Payment mode selection triggers validation and UI updates for EMI information.
- Calculate Premium button invokes premium calculation including GST via a stored procedure.
- OTP is sent via loyalty package and user confirmation validates OTP against logs.
- Document upload is managed in a separate block with file upload integration through WebUtil.
- Minimum 4 documents are required before allowing save; upload errors are displayed.
- Addon package premium calculations are performed on demand and displayed.
- Commit process validates data, commits to DB, generates reports, and handles errors.
- Exit clears session info and closes the form.
- Error handling and user messages are managed through the GENERAL package and alerts.
- WebUtil bean areas handle client info, file, session, and other utility functions with custom event handlers.

Document assembled from gm\_ewarr\_sale\_t2 functional sections, comprehensive technical documentation, and Code Snippets references – © 2025 Happiestminds Technologies.