Home360 - Unified Home Solution

Problem Statement

Managing household needs such as Maintenance issues, Property dues, event reservations etc is often a cumbersome and disjointed process for homeowners. Existing solutions lack a unified platform to address these diverse needs efficiently. Additionally, these solutions often do not leverage advanced technologies like GenAI for predictive maintenance, AI-driven task automation, or robust document and payment management. Homeowners need an integrated application that simplifies household management tasks while incorporating the latest technological advancements.

Solution

Home360, powered by Salesforce, addresses these challenges by providing a comprehensive platform designed specifically for homeowners. The application offers a range of features, including:
Predictive Maintenance: Utilizing GenAI to predict and address maintenance issues before they become major problems.
AI-Driven Task Automation: Streamlining routine tasks and reminders to enhance efficiency.
Document Management: Centralizing important documents for easy access and organization.
Salesforce Payment Process: Facilitating property dues and other payments through a secure and user-friendly interface that can reflect and regenerate transactions efficiently.
Event Reservations: Simplifying the booking process for events such as birthday parties and community gatherings.
Moreover, Home360 is not just limited to property management. This plug-and-play solution can be extended to different areas, providing versatile and scalable applications for a variety of household and community needs.

Advantages of Using Salesforce Home360

1. **Unified Platform**: Home360 combines various household management tasks into one cohesive application, reducing the need for multiple disparate tools.
2. **Advanced Technology**: Leveraging GenAI and AI-driven automation enhances the efficiency and predictive capabilities of household management.
3. **Centralized Document Management**: Easy access and organization of important documents ensure homeowners can quickly find what they need.
4. **Secure Payment Processing**: Built-in payment features provide a secure and straightforward way to handle financial transactions, reflecting and regenerating them as needed.
5. **Simplified Event Management**: Streamlined event reservation processes save time and reduce hassle for homeowners.
6. **Extendable Solutions**: Beyond property management, Hom360 can be adapted to various household and community needs, offering versatile plug-and-play solutions.
7. **Salesforce Integration**: Harnessing the power of the Salesforce platform ensures reliability, scalability, and robust support.

By adopting Salesforce Home360, homeowners can experience a more organized, efficient, and technologically empowered approach to managing their household needs, with the flexibility to extend its benefits to various aspects of their daily lives.

Introduction

This document outlines the design, data model, and process flow for the Home Depot (HD) and Homeowner Association (HOA) Salesforce app. It includes key processes, a detailed data model, and the potential use of Generative AI (GenAI) to streamline operations.

KEY PROCESSES IN HOA MANAGEMENT

Membership Management

- Onboarding new members
- Managing member information
- Handling member payments and dues

Property Management

- Tracking properties within the association
- Managing property-related documents and compliance

Financial Management

- Managing budgets and expenses
- Handling member dues and payments
- Generating financial reports

Communication

- Sending notifications and updates to members
- Managing communication channels (email, SMS, etc.)

Complaint and Request Management

- Logging and tracking member complaints and requests
- Assigning tasks to appropriate personnel
- Monitoring resolution status

Event Management

- Planning and managing community events
- Sending event notifications and reminders
- Tracking event participation

Directory Management

- Adding Board Members
- Updating Board Members
- Adding/updating phone numbers and email ID

Document Management

- Adding Documents
- Updating Documents

Data Model

~~Member~~/Account

- ~~MemberID (Primary Key)~~
- FirstName
- LastName
- Email
- Phone

- Address
- IsBoardMember
- MembershipStartDate
- MembershipEndDate
  Note : May not be required below two and can use Financials entity
- DuesPaid (Boolean)
- DuesAmount

Property (HD and HOA App)

- ~~PropertyID (Primary Key)~~
- Address
- OwnerID (Foreign Key to Member) (AccountId)
- PropertyType
- ComplianceStatus
- Documents (File Links)

Financials (HD and HOA App)

- TransactionID (Primary Key). **Id (Will refer to default recordId)**
- MemberID (Foreign Key to Member) **(Referring to Account recordID)**
- Amount
- TransactionType (Fine, Payment, Others)
- **InitiationDate**
- TransactionDate
- Description
- **Status (Paid/Delinquent)**

Communication (HD and HOA App)

- CommunicationID (Primary Key) **Id (Will refer to default recordId)**
- MemberID (Foreign Key to Member)
- CommunicationType (Email, SMS, Mail, Call)
- Subject
- Message
- SentDate

Complaint/Request

- RequestID (Primary Key) **Id (Will refer to default recordId)**
- MemberID (Foreign Key to Member)
- RequestType (Complaint, Service Request)
- Description
- Status (Open, In Progress, Closed)
- AssignedTo
- CreatedDate

- ClosedDate

Event

- EventID (Primary Key) Id (Will refer to default recordId)
- EventName
- Description
- EventDate
- Location
- OrganizerID (Foreign Key to Account)
- Participants (Related List to Account) (May not be required , based on requirement we can change)

Where to Use Generative AI (GenAI)

- Member Onboarding: GenAI can help streamline the onboarding process by generating personalized welcome emails, membership contracts, and other necessary documentation.
- Communication: Use GenAI to draft emails, SMS, and notices. For example, based on the context, GenAI can draft reminders for due payments, announcements for upcoming events, or responses to common queries.
- Complaint and Request Management: Implement a chatbot powered by GenAI to handle initial member complaints and requests. The chatbot can provide quick responses, gather necessary details, and even suggest solutions based on historical data.
- Event Management: GenAI can help in creating event invitations, reminders, and follow-up messages. It can also assist in generating summaries of event participation and feedback.
- Financial Reports: Use GenAI to generate detailed financial reports and summaries. GenAI can analyze transaction data and provide insights into the financial health of the HOA.

Salesforce Implementation

- Custom Objects: Define custom objects for Members, Properties, Financials, Communication, Complaints/Requests, and Events.
- Automation: Use Salesforce Flow to automate processes such as sending reminders, updating statuses, and generating reports.
- Email Templates: Create email templates for various communication needs, leveraging GenAI for content generation.
- Reports and Dashboards: Use Salesforce's reporting tools to create real-time dashboards and reports for financials, member activities, and property management.
- Einstein AI: Integrate Salesforce Einstein to leverage AI capabilities for predictive analytics, especially in financial forecasting and member engagement.

Entity-Relationship Diagram (ERD)

TODO

Membership Management Flow

[New Member Request] → [Verification] → [Create Member Record] → [Send Welcome Email (GenAI)] → [Membership Active]

Property Management Flow

[Add Property] → [Assign Owner] → [Upload Documents] → [Track Compliance]

Financial Management Flow

[Generate Invoice] → [Send Invoice (GenAI)] → [Receive Payment] → [Update Financial Record] → [Generate Financial Report

(GenAI)]

Communication Flow

[Create Communication] → [Draft Message (GenAI)] → [Send Message] → [Track Responses]

Complaint and Request Management Flow

[Receive Complaint/Request] → [Assign to Personnel] → [Update Status] → [Send Updates to Member (GenAI)] → [Close Complaint/Request]

Event Management Flow

[Create Event] → [Draft Invitations (GenAI)] → [Send Invitations] → [Track RSVPs] → [Manage Event] → [Send Follow-up (GenAI)]

Applications Flow

1. Membership Management Flow with DB Updates

- **New Member Request**
  - **Description**: A new member requests to join the association.
  - **Process**:
    i. Member submits a new member request.
    ii. Request undergoes verification.
- **Verification**
  - **Description**: Verify the details of the new member request.
  - **Process**:
    i. Verify member details and eligibility.
    ii. Validate membership criteria.
- **Create Member Record**
  - **Description**: Create a record for the new member in the system.
  - **Process**:
    i. Record member's information in the `Member` table.
    ii. Assign MemberID and other necessary details.
- **Send Welcome Email (GenAI)**
  - **Description**: Automatically send a welcome email to the new member.
  - **Process**:
    i. Generate a personalized welcome email.
    ii. Send email using automated tools.
- **Membership Active**
  - **Description**: Activate membership status for the new member.
  - **Process**:
    i. Update membership status in the `Member` table to "Active".
    ii. Member can now participate in association activities.

2. Property Management Flow

- **Add Property**
  - **Description**: Add a new property to the association's management system.
  - **Process**:
    i. Enter property details into the `Property` table.
    ii. Assign PropertyID and other relevant attributes.
- **Assign Owner**
  - **Description**: Assign an owner to a property within the association.
  - **Process**:
    i. Link member (owner) to the property using MemberID and PropertyID.
- **Upload Documents**
  - **Description**: Upload documents related to property management.
  - **Process**:
    i. Upload documents and link them to the respective property in the `Property` table.
    ii. Manage document versions and access permissions.
- **Track Compliance**
  - **Description**: Monitor compliance status for properties.
  - **Process**:
    i. Track adherence to association rules and regulations.
    ii. Update compliance status in the `Property` table.

3. Financial Management Flow

- **Generate Invoice**
  - **Description**: Create an invoice for member dues or other financial obligations.
  - **Process**:
    i. Generate invoice details in the `Financials` table.
    ii. Include MemberID, Amount, TransactionType (e.g., Dues), and TransactionDate.
- **Send Invoice (GenAI)**
  - **Description**: Automatically send generated invoices to members.
  - **Process**:
    i. Use automated tools to send invoices via email or other communication channels.
- **Receive Payment**
  - **Description**: Record member payments against generated invoices.
  - **Process**:
    i. Update payment details in the `Financials` table.
    ii. Mark invoices as paid.
- **Update Financial Record**

- **Description**: Maintain accurate financial records for transactions.
- **Process**:
    i. Log financial transactions (payments, dues, expenses) in the `Financials` table.

- **Generate Financial Report (GenAI)**
  - **Description**: Automatically generate financial reports based on transaction data.
  - **Process**:
    i. Use data analytics tools to compile financial information.
    ii. Generate reports summarizing income, expenses, and financial trends.

4. Communication Flow

- **Create Communication**
  - **Description**: Initiate communication with association members.
  - **Process**:
    i. Create communication records in the `Communication` table.
    ii. Include MemberID, CommunicationType (Email, SMS, Notice), Subject, Message, and SentDate.
- **Draft Message (GenAI)**
  - **Description**: Draft messages automatically using AI tools.
  - **Process**:
    i. Generate message content based on predefined templates or member data.
- **Send Message**
  - **Description**: Send messages to members via selected communication channels.
  - **Process**:
    i. Use automated tools to deliver messages promptly.
- **Track Responses**
  - **Description**: Monitor member responses and engagement.
  - **Process**:
    i. Log responses and interaction details in the `Communication` table.

5. Complaint and Request Management Flow

- **Receive Complaint/Request**
  - **Description**: Receive and log member complaints or service requests.
  - **Process**:
    i. Record complaints or requests in the `ComplaintRequest` table.
    ii. Include MemberID, RequestType (Complaint, Service Request), Description, AssignedTo, CreatedDate.
- **Assign to Personnel**
  - **Description**: Assign complaints or requests to appropriate personnel for resolution.
  - **Process**:

        i. Allocate tasks to designated personnel.
        ii. Update status and assignment details.

- **Update Status**
    - **Description**: Track the progress and status of complaints or requests.
    - **Process**:
        i. Update status (Open, In Progress, Closed) in the `ComplaintRequest` table.
        ii. Record any actions taken or resolutions achieved.
- **Send Updates to Member (GenAI)**
    - **Description**: Automatically notify members of complaint/request updates.
    - **Process**:
        i. Use automated messaging tools to inform members.
- **Close Complaint/Request**
    - **Description**: Mark complaints or requests as resolved or closed.
    - **Process**:
        i. Update status to "Closed" in the `ComplaintRequest` table.
        ii. Finalize any necessary documentation.

6. Event Management Flow

- **Create Event**
    - **Description**: Plan and schedule community events.
    - **Process**:
        i. Enter event details in the `Event` table.
        ii. Include EventName, Description, EventDate, Location, OrganizerID.
- **Draft Invitations (GenAI)**
    - **Description**: Automatically draft event invitations.
    - **Process**:
        i. Generate personalized invitations using AI tools.
- **Send Invitations**
    - **Description**: Distribute event invitations to members.
    - **Process**:
        i. Send invitations via email or other communication channels.
- **Track RSVPs**
    - **Description**: Monitor member responses to event invitations.
    - **Process**:
        i. Record RSVPs and attendance details in the `EventParticipants` table.
- **Manage Event**

- **Description**: Coordinate logistics and activities during the event.
- **Process**:
    i. Ensure smooth execution and participant satisfaction.

- **Send Follow-up (GenAI)**
    - **Description**: Automatically send event follow-up messages.
    - **Process**:
        i. Use automated tools to thank attendees and gather feedback.

SQL Script

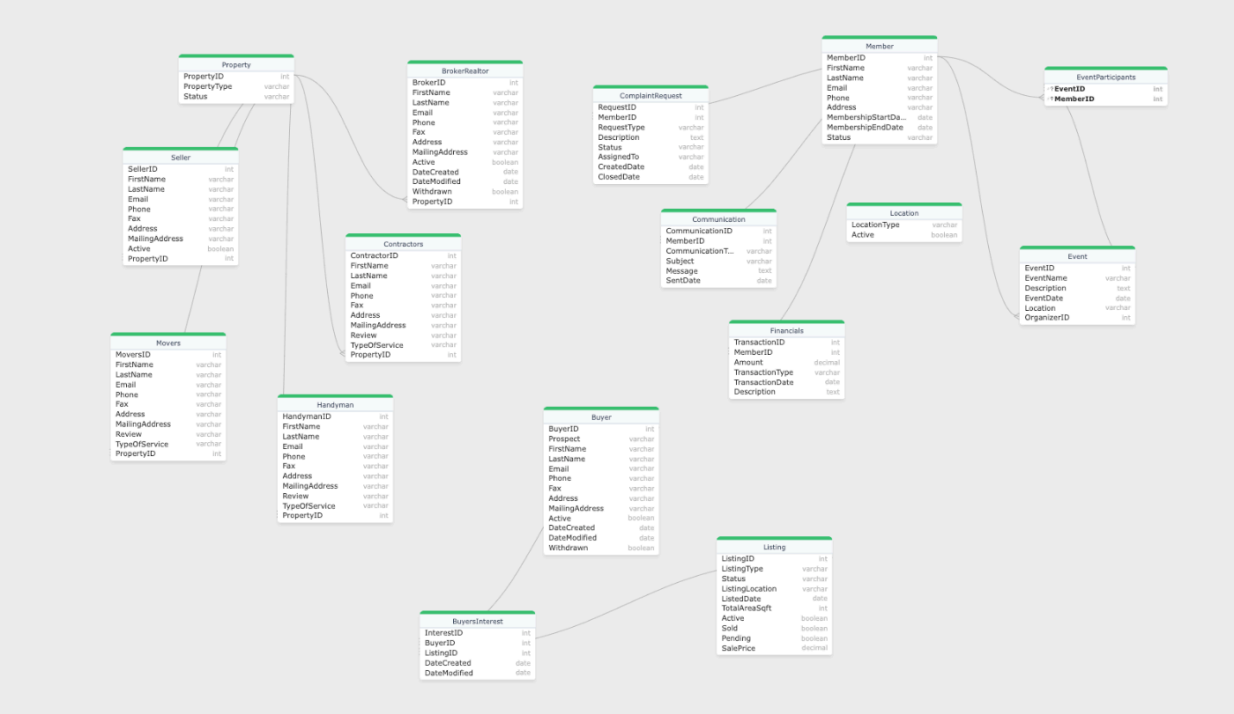
SQL Script

```
-- Tables for Homeowner Association (HOA)
CREATE TABLE Member (
    MemberID INT PRIMARY KEY,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    Email VARCHAR(100),
    Phone VARCHAR(20),
    Address VARCHAR(255),
    MembershipStartDate DATE,
    MembershipEndDate DATE,
    Status VARCHAR(50)
);

CREATE TABLE Property (
    PropertyID INT PRIMARY KEY,
    PropertyType VARCHAR(50),
    Status VARCHAR(50) -- Active/Inactive
);

CREATE TABLE Financials (
    TransactionID INT PRIMARY KEY,
    MemberID INT,
    Amount DECIMAL(10, 2),
    TransactionType VARCHAR(50),
    TransactionDate DATE,
    Description TEXT,
    FOREIGN KEY (MemberID) REFERENCES Member(MemberID)
);

CREATE TABLE Communication (
    CommunicationID INT PRIMARY KEY,
    MemberID INT,
    CommunicationType VARCHAR(50),
    Subject VARCHAR(100),
    Message TEXT,
    SentDate DATE,
    FOREIGN KEY (MemberID) REFERENCES Member(MemberID)
);
```

```
CREATE TABLE ComplaintRequest (
    RequestID INT PRIMARY KEY,
    MemberID INT,
    RequestType VARCHAR(50),
    Description TEXT,
    Status VARCHAR(50),
    AssignedTo VARCHAR(50),
    CreatedDate DATE,
    ClosedDate DATE,
    FOREIGN KEY (MemberID) REFERENCES Member(MemberID)
);

CREATE TABLE Event (
    EventID INT PRIMARY KEY,
    EventName VARCHAR(100),
    Description TEXT,
    EventDate DATE,
    Location VARCHAR(255),
    OrganizerID INT,
    FOREIGN KEY (OrganizerID) REFERENCES Member(MemberID)
);

CREATE TABLE EventParticipants (
    EventID INT,
    MemberID INT,
    PRIMARY KEY (EventID, MemberID),
    FOREIGN KEY (EventID) REFERENCES Event(EventID),
    FOREIGN KEY (MemberID) REFERENCES Member(MemberID)
);

-- Tables for Home Depot Buyer
CREATE TABLE Buyer (
    BuyerID INT PRIMARY KEY,
    Prospect VARCHAR(50),
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    Email VARCHAR(100),
    Phone VARCHAR(20),
    Fax VARCHAR(20),
    Address VARCHAR(255),
    MailingAddress VARCHAR(255),
    Active BOOLEAN,
    DateCreated DATE,
    DateModified DATE,
    Withdrawn BOOLEAN
);

CREATE TABLE BuyersInterest (
    InterestID INT PRIMARY KEY,
    BuyerID INT,
    ListingID INT,
    DateCreated DATE,
    DateModified DATE,
    FOREIGN KEY (BuyerID) REFERENCES Buyer(BuyerID),
```

```
    FOREIGN KEY (ListingID) REFERENCES Listing(ListingID)
);

CREATE TABLE Location (
    LocationType VARCHAR(50),
    Active BOOLEAN
);

CREATE TABLE Listing (
    ListingID INT PRIMARY KEY,
    ListingType VARCHAR(50),
    Status VARCHAR(50), -- Active/Inactive
    ListingLocation VARCHAR(255),
    ListedDate DATE,
    TotalAreaSqft INT,
    Active BOOLEAN,
    Sold BOOLEAN,
    Pending BOOLEAN,
    SalePrice DECIMAL(10, 2)
);

CREATE TABLE Seller (
    SellerID INT PRIMARY KEY,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    Email VARCHAR(100),
    Phone VARCHAR(20),
    Fax VARCHAR(20),
    Address VARCHAR(255),
    MailingAddress VARCHAR(255),
    Active BOOLEAN,
    PropertyID INT,
    FOREIGN KEY (PropertyID) REFERENCES Property(PropertyID)
);

CREATE TABLE BrokerRealtor (
    BrokerID INT PRIMARY KEY,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    Email VARCHAR(100),
    Phone VARCHAR(20),
    Fax VARCHAR(20),
    Address VARCHAR(255),
    MailingAddress VARCHAR(255),
    Active BOOLEAN,
    DateCreated DATE,
    DateModified DATE,
    Withdrawn BOOLEAN,
    PropertyID INT,
    FOREIGN KEY (PropertyID) REFERENCES Property(PropertyID)
);

CREATE TABLE Contractors (
    ContractorID INT PRIMARY KEY,
    FirstName VARCHAR(50),
```

```
    LastName VARCHAR(50),
    Email VARCHAR(100),
    Phone VARCHAR(20),
    Fax VARCHAR(20),
    Address VARCHAR(255),
    MailingAddress VARCHAR(255),
    Review VARCHAR(255),
    TypeOfService VARCHAR(255),
    PropertyID INT,
    FOREIGN KEY (PropertyID) REFERENCES Property(PropertyID)
);

CREATE TABLE Handyman (
    HandymanID INT PRIMARY KEY,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    Email VARCHAR(100),
    Phone VARCHAR(20),
    Fax VARCHAR(20),
    Address VARCHAR(255),
    MailingAddress VARCHAR(255),
    Review VARCHAR(255),
    TypeOfService VARCHAR(255),
    PropertyID INT,
    FOREIGN KEY (PropertyID) REFERENCES Property(PropertyID)
);

CREATE TABLE Movers (
    MoversID INT PRIMARY KEY,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
    Email VARCHAR(100),
    Phone VARCHAR(20),
    Fax VARCHAR(20),
    Address VARCHAR(255),
    MailingAddress VARCHAR(255),
    Review VARCHAR(255),
    TypeOfService VARCHAR(255),
    PropertyID INT,
    FOREIGN KEY (PropertyID) REFERENCES Property(PropertyID)
);

-- Establishing Relationships for BuyersInterest
ALTER TABLE BuyersInterest
    ADD COLUMN BuyerID INT,
    ADD FOREIGN KEY (BuyerID) REFERENCES Buyer(BuyerID);

ALTER TABLE BuyersInterest
    ADD COLUMN ListingID INT,
    ADD FOREIGN KEY (ListingID) REFERENCES Listing(ListingID);

-- Adding PropertyID to Location
ALTER TABLE Location
    ADD COLUMN PropertyID INT,
    ADD FOREIGN KEY (PropertyID) REFERENCES Property(PropertyID);
```



Disclaimer:
This is a working document which will provide the roadmap for Home360 that we are planning to extend and enhance to give end to end solution to all home owners using GenAI.