BLUEPRINT

UML – CSP 586

TEAM #24

Agarwal, Vivek (A20384303)
Bondugula, Puneeth Reddy (A20379964)
Meghana Subbanna, FNU(A20381262)
Namachivayan, Sivasenthil (A20391478)
Puttaraju, Ravikiran (A20387024)

TABLE OF CONTENTS

HOUSING DEVELOPMENT DASHBOARD APPLICATION

1. PROJECT OVERVIEW:

A **housing development** is a structured real estate development of residential buildings. Popular throughout the United States and the United Kingdom, they are often areas of high-density, low-impact residences of single-family detached homes, and often allow for separate ownership of each housing unit, for example through subdivision. A housing development is *"often erected on a tract of land by one builder and controlled by one management."*

Housing development system application is used to display the development of the housing finance, affordable rental housing, affordable housing inventory, building permits and vacant residential building.

The system used the data provided by the various data portal for displaying the different housing development activities. The Dashboard Data Portal is an object oriented charting Dashboard for data analytics and visualization for datasets in a Housing Development segment of five cities. The site hosts over 5 datasets presented in easy-to-use formats about housing development related to following:

- Austin: Affordable Housing Inventory.
- Chicago: Affordable rental housing.
- Connecticut: Vaccant residential building.
- NewYork: Housing finance, Maximum affordable units.
- Seattle: Housing and building permits.

The housing development system allows the user to upload the dataset and gives a graphical representation of data i, e chart, in which symbols represent the data, such as bars in a bar chart, lines in a line chart, or slices in a pie chart, donoughnut chart and stacked chart. A chart can represent tabular numeric data, functions or some kinds of qualitative structure and provides different info.

A **data chart** is a type of diagram or graph, that organizes and represents a set of numerical or qualitative data. Certain types of charts are more useful for presenting a given data set than others. For example, data that presents percentages in different groups (such as "Housing finance, regular, unsure") are often displayed in a pie chart, but may be more easily understood when presented in a horizontal bar chart. On the other hand, data that represents numbers that change over a period (such as "annual Housing finance from 2010 to 2016") might be best shown as a line chart.

This system also gives an option to choose to invoke housing updates, to select number of dwelling units, to select permitted housing units, display statistics and to share property valuation.

2. REQUIREMENTS AND FEATURES:

System should be able to display the housing development statistics such as housing finance, affordable rental housing, affordable housing inventory, building permits and Vacant Residential Buildings of five different cities of United States.

i. User has the option to select the dataset from .csv file to read it in table.

ii. User will be provided with 5 different links as a drop down to select the dataset.

iii. Once the user selects any dataset, the data will be loaded and displayed as a table in the dashboard (viewed through browser).

iv. User can perform the search operation by entering the keyword of the data.

v. User can sort the column by ascending/descending order or alphabetical order.

vi. User can filter the data by selecting the column names and then by numerical values like greater than, lesser than or equal to.

vii. User can select the graph/chart which he wants to plot out of the five different charts like bar graph, line graph, pie chart, donought chart or stacked chart.

viii. The user can select/deselect Columns for inclusion/exclusion of data from the dataset for Charting the selected data

ix. The user can select/deselect Columns for inclusion/exclusion of selected data from the dataset for charting.

x. The user can filter Rows for inclusion/exclusion of the data from the dataset for charting

xi. Able to view the building permits, permit type, category, action type. Etc.

xii. User has the option to view the Affordable Housing data of Austin for different Management company, Development Name.

xiii. User can select the Building permit data of Seattle and filter it based on the Permit type, Action Type, Work Type, Value, and status.

xiv. Building permits for already constructed building, a field inspection is mandatory. In this case, a Plan review is not required. Whereas for a new building there is no field inspection is required but a plan review is required and the application must be accepted. Whenever a field inspection is done, the department must issue permit.

xv. For demolition, a field inspection is necessary and there is no need of plan review. Successful review must be completed for successful demolition.

XVI. The property valuation data can be viewed by selecting full value of the property, zip code, year.

XVII. The user can create and plot/chart complex analytical queries based on the column names and values for a given data set like greater than, lesser than, equal to.

3. USE CASE:

3.1. ACTORS

1. User 2. System

Sl. No	Name (Actors)	Role
01	User	▪ User can import the dataset. ▪ User can filter the rows ▪ User can select the columns ▪ User can view the chart ▪ User can search ▪ User can sort the data
02	System	▪ This is the system comprising all information.

3.2. USE CASE DESCRIPTION

Use Case Id	Use Case Name	Description
1	ChooseDataset	▪ User can load the dataset from either .csv file and convert it into Json.
2	SelectDataset	▪ User will select the dataset out of the 5 available Housing development dataset.
3	DataSelection	▪ User can select the columns to plot.
4	SelectChartType	▪ User can plot the chart based from the list. ▪ Bar Chart ▪ Pie Chart ▪ Line Chart ▪ Donought Chart ▪ Stacked Chart
5	DataFIltering	▪ User can filter the rows from the dataset to plot the graph.
6	Execute Basic Statistical Metrics for Numerical Columns	▪ To display the statistical data for the selected columns. ▪ Equal to, lesser than, greater than
7	Execute SQL queries on dataframes	▪ Queries are performed on the selected dataframe

3.3. USE CASE DIAGRAM

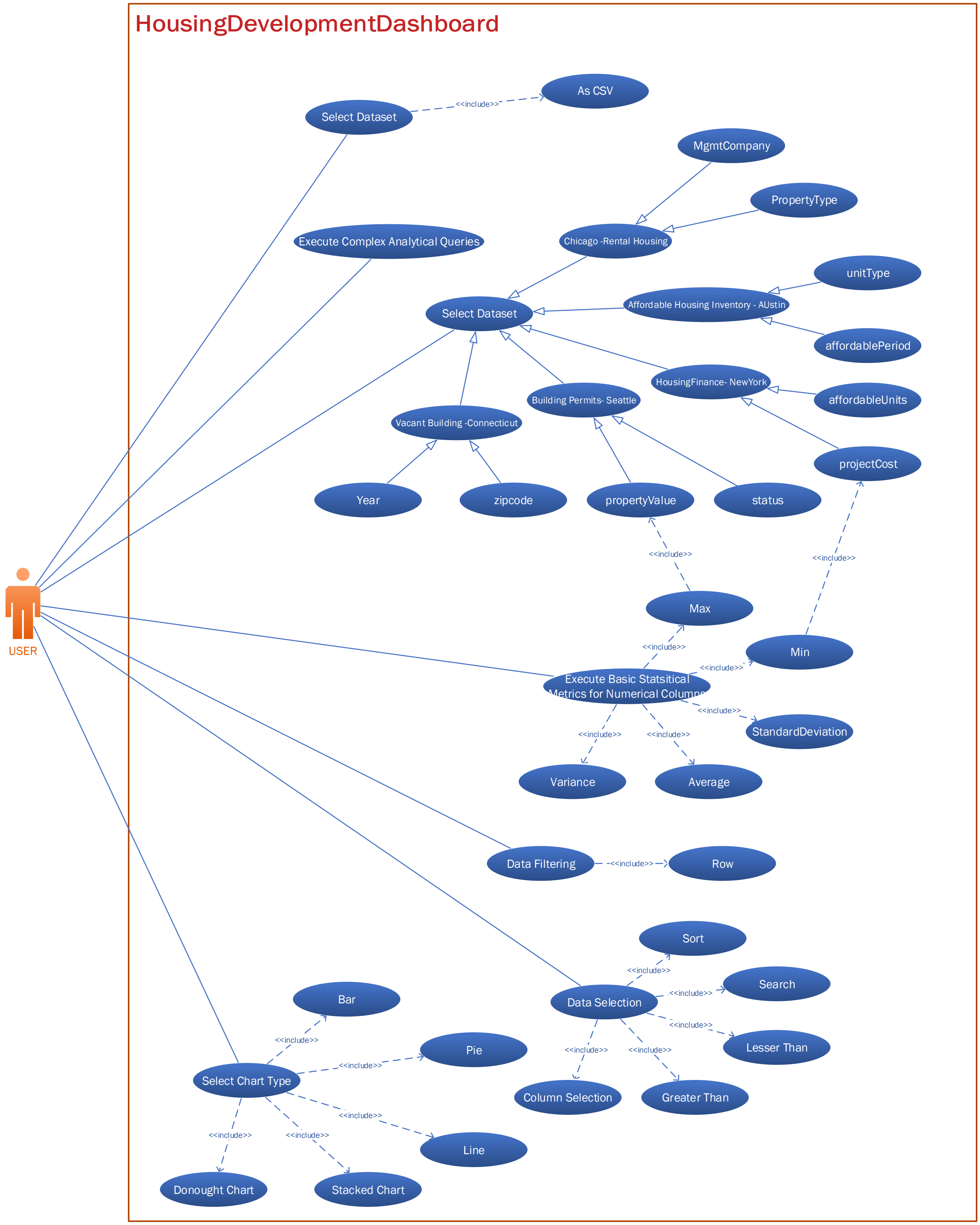


Fig: Use Case Diagram of charting Dashboard

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements.

In this context, the term "system" refers to something being developed or operated in our project Housing Development dashboard system is being developed.

In our diagram, End User in a primary actor, who interacts with the system directly. The user will select the different dataset and view it in the dashboard. If he wants to apply some filter he can do so by selecting a range of values, date. He can do the calculations based on the statistical values.

The user can plot the chart for the selected range of data. He can plot any of the five graph. If he wishes to apply the data-filtering and data-selection further, the same will be reflected in the graph.

The Dashboard Data Portal is an object oriented charting Dashboard for data analytics and visualization for datasets in a Housing Development segment of five cities. The site hosts over 5 datasets presented in easy-to-use formats about housing finance, affordable rental housing, affordable housing inventory, building permits and vacant residential building.

4. FULLY DRESSED USE CASE

<table>
<tr><th>Use case Name</th><th colspan="2">To display Chart</th></tr>
<tr><td><b>Scenario</b></td><td colspan="2">To display the chart.</td></tr>
<tr><td><b>Triggering Event</b></td><td colspan="2">User viewing the chart (Pictorial data) as per his selection</td></tr>
<tr><td><b>Brief Description</b></td><td colspan="2">The user make sure that the columns selected have the appropriate values to be plotted as a chart.</td></tr>
<tr><td><b>Actors</b></td><td colspan="2">User</td></tr>
<tr><td><b>Stakeholders</b></td><td colspan="2">User</td></tr>
<tr><td><b>Related Use Case</b></td><td colspan="2">Includes: Data Selection, Data Filtering</td></tr>
<tr><td><b>Preconditions</b></td><td colspan="2">The user must select only the proper fields for plotting the chart.
The selected column should have the valid data.
The range of data should be within limit.</td></tr>
<tr><td><b>Post Condition</b></td><td colspan="2">User is eligible for analysing the dataset in the statistical way.</td></tr>
<tr><td><b>Success Guarantee</b></td><td colspan="2">▪ IF the validation passes without any error.
▪ The selected value should be in the range to display.
▪ A valid output should be displayed.</td></tr>
<tr><td rowspan="4"><b>Flow of events</b></td><td><b>Actors</b></td><td><b>System</b></td></tr>
<tr><td>Import the data set</td><td>Display the dataset</td></tr>
<tr><td>Select the columns which needed to be plotted in the chart</td><td>Verify whether the appropriate column is selected</td></tr>
<tr><td>Display the chart</td><td></td></tr>
<tr><td><b>Exception condition</b></td><td colspan="2">Unable to select the dataset.
Selecting the inappropriate columns for the line chart
Dynamic allocation of the X axis and Y axis values does not happen
If there is no data in the selected range</td></tr>
</table>

4.1. Fully dressed format of Use Case Plotting Line Chart:

Use case Name	**Plot Line Chart for Newyork dataset**	
Scenario	Plot Line chart between Developer name and Total Units	
Triggering Event	User viewing the chart (Pictorial data) as per his selection	
Brief Description	The user make sure that the columns selected have the appropriate values to be plotted as a line chart. The user needs to select the total units (numeric data) for Y axis and Developer Name for the X axis	
Actors	User	
Stakeholders	User	
Related Use Case	Includes: Data Selection, Data Filtering	
Preconditions	The user must select only the proper fields for plotting the line chart	
Post Condition	User is eligible for analysing the dataset in the statistical way	
Flow of events	**Actors**	**System**
	Import the data set	Display the dataset
	Select the columns which needed to be plotted in the chart	Verify whether the appropriate column is selected
	Display the chart	
Exception condition	unable to select the dataset Selecting the inappropriate columns for the line chart Dynamic allocation of the X axis and Y axis values does not happen If there is no data in the selected range	

4.2. Fully dressed format of Use Case for Greater value in the dataset:

Use case Name	**Determine Greater value of a column for a dataset**	
Scenario	Determine the Greater value of Affordable units column from the dataset	
Triggering Event	User viewing the Greater value as per his selection	
Brief Description	The user make sure that the columns selected have the appropriate numeric values to be identified with the Greater AX Value. The user needs to select the column and invoke calculate max to find the max value in that particular column	
Actors	User	
Stakeholders	Includes: Data Selection, Data Filtering	
Related Use Case	User	
Preconditions	The user must select only the proper numeric field to find the Max column	
Post Condition	User is eligible for analyzing the dataset in the statistical way	
Flow of events	**Actors**	**System**
	Import the data set	Display the dataset
	Select the columns which needed to be plotted in the chart. Select than Greater than radio button. Enter the range of number.	Verify whether the appropriate column is selected
	Display the chart	
Exception condition	unable to select the dataset Selecting the inappropriate columns for the line chart	

4.3. Fully dressed format of Use Case for Lesser than value of dataset - Connecticut:

Use case Name	**Determine lesser than value of a column for a dataset**	
Scenario	Plot Pie Chart for Connecticut dataset	
Triggering Event	Plot Pie Chart based on the total number of units for a particular Year	
Brief Description	User viewing the chart(Pictorial data) as per his selection	
Actors	The user make sure that the columns selected have the appropriate values to be plotted as a pie chart. The user needs to select the total units (numeric data) for Y axis and Property Type for the X axis	
Stakeholders	User	
Related Use Case	User	
Preconditions	Includes: Data Selection, Data Filtering	
Post Condition	The user must select only the required fields for plotting the pie chart	
Flow of events	**Actors**	**System**
	Import the data set	Display the dataset
	Select the columns which needed to be plotted in the chart	Verify whether the appropriate column is selected
	Display the chart	Compute the total number of units for a particular Year
Exception condition	unable to select the dataset Selecting the inappropriate columns for the line chart If the Dynamic allocation of the values does not happen	

4.4. Fully dressed format of Use Case for Determine Equal value for data available in column for a dataset.

Use case Name	**Determine Equal value for data available in column for a dataset**	
Scenario	Determine the Equal unit value for a particular area based on unit value	
Triggering Event	User viewing the equal unit value as per his selection	
Brief Description	The user make sure that the columns selected have the appropriate numeric values to calculate the equivalent value. The user needs to select the column and invoke calculate equal to find the equal value in that particular column	
Actors	User	
Stakeholders	Includes: Data Selection, Data Filtering	
Related Use Case	User	
Preconditions	The user must select only the proper numeric value to find the equivalent number in the graph.	
Post Condition	User is eligible for view the chart for the selected value.	
Flow of events	**Actors**	**System**
	Import the data set	Display the dataset
	Select the columns which needed to be plotted in the chart	Verify whether the appropriate column is selected
	Display the chart	Compute the total number of units for a particular Year
Exception condition	unable to select the dataset Selecting the inappropriate columns for the line chart Dynamic allocation of the X axis and Y axis values does not happen	

4.5. Fully dressed format of Use Case for Plotting Bar Graph for Chicago dataset

Use case Name	**Plot Bar Graph for Chicago dataset**	
Scenario	Plot Bar Graph based on the total number of units for a particular Property Type.	
Triggering Event	User viewing the chart(Pictorial data) as per his selection	
Brief Description	The user make sure that the columns selected have the appropriate values to be plotted as a line chart. The user needs to select the total units (numeric data) for Y axis and Property Type for the X axis	
Actors	User	
Stakeholders	User	
Related Use Case	Includes: Data Selection, Data Filtering	
Preconditions	The user must select only the required fields for plotting the line chart	
Post Condition	User is eligible for analyzing the dataset in the statistical way	
Flow of events	**Actors**	**System**
	Import the data set	Display the dataset
	Select the columns which needed to be plotted in the chart	Verify whether the appropriate column is selected
	Display the chart	Compute the total number of units for a particular Property Type
Exception condition	Display the chart Unable to select the dataset Selecting the inappropriate columns for the line chart Dynamic allocation of the X axis and Y axis values does not happen	

4.6. Fully dressed format of Use Case to Determine Lesser value of a column for a dataset

Use case Name	Determine Lesser value of a column for a dataset	
Scenario	Determine the Lesser unit value for a Management Company	
Triggering Event	User viewing the Lesser value as per his selection	
Brief Description	The user make sure that the columns selected have the appropriate numeric values to be identified with the least value. The user needs to select the column and invoke calculate min to find the least value in that column.	
Actors	User	
Stakeholders	Includes: Data Selection, Data Filtering	
Related Use Case	User	
Preconditions	The user must select only the proper numeric field to find the lesser value in the column	
Post Condition	User is eligible for analyzing the dataset in the statistical way	
Flow of events	**Actors**	**System**
	Import the data set	Display the dataset
	Select the numeric column for which the min value is to be determined	Verify whether the appropriate column is selected
	Display the lesser value selected for a range	
Exception condition	Display the chart Unable to select the dataset Selecting the inappropriate columns for the line chart	

4.7. Fully dressed format of Use Case to Plot Line Chart for Building Permits dataset

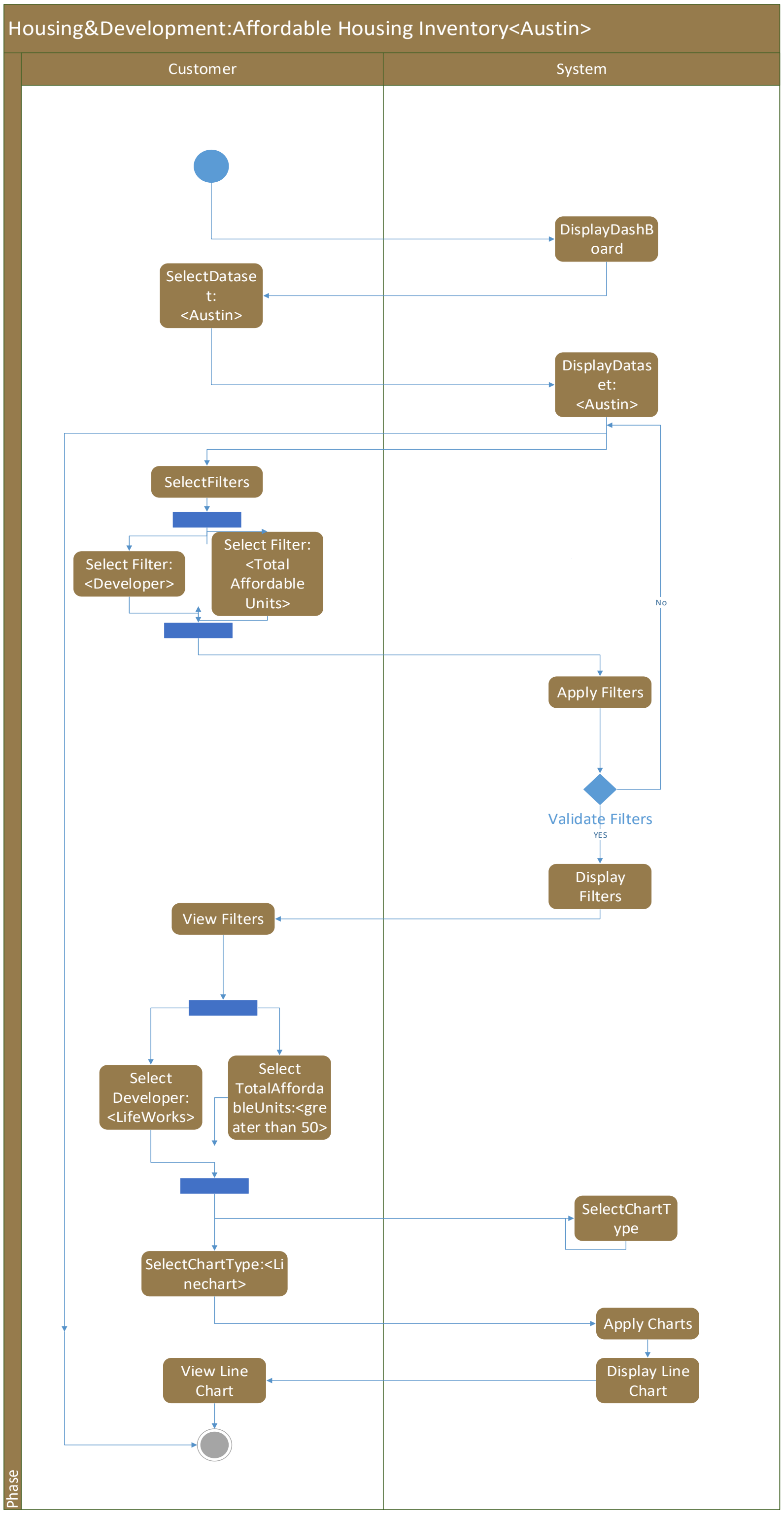
Use case Name	Plot Line Chart for Building Permits dataset	
Scenario	Plot Line chart between PermitType and Total Value	
Triggering Event	User viewing the chart(Pictorial data) as per his selection	
Brief Description	The user make sure that the columns selected have the appropriate values to be plotted as a line chart. The user needs to select the total Value(numeric data) for Y axis and PermitType for the X axis	
Actors	User	
Stakeholders	User	
Related Use Case	Includes: Data Selection, Data Filtering	
Preconditions	The user must select only the proper fields for plotting the line chart	
Post Condition	User is eligible for analyzing the dataset in the statistical way	
Flow of events	**Actors**	**System**
	Import the Building Permit as data set	Display the Building Permit dataset
	Select the columns which needed to be plotted in the chart	Verify whether the appropriate column is selected
	Display the chart	
Exception condition	unable to select the dataset Selecting the inappropriate columns for the line chart Dynamic allocation of the X axis and Y axis values does not happen	

4.8. Fully dressed format of Use Case to draw donought chart for a dataset.

<table>
<tr><td><b>Use case Name</b></td><td colspan="2"><b>To display Donought Chart</b></td></tr>
<tr><td><b>Scenario</b></td><td colspan="2">To display the donought chart.</td></tr>
<tr><td><b>Triggering Event</b></td><td colspan="2">User viewing the chart (Pictorial data) as per his selection</td></tr>
<tr><td><b>Brief Description</b></td><td colspan="2">The user make sure that the columns selected have the appropriate values to be plotted as a chart.</td></tr>
<tr><td><b>Actors</b></td><td colspan="2">User</td></tr>
<tr><td><b>Stakeholders</b></td><td colspan="2">User</td></tr>
<tr><td><b>Related Use Case</b></td><td colspan="2">Includes: Data Selection, Data Filtering</td></tr>
<tr><td><b>Preconditions</b></td><td colspan="2">The user must select only the proper fields for plotting the chart.
The selected column should have the valid data.
The range of data should be within limit.</td></tr>
<tr><td><b>Post Condition</b></td><td colspan="2">User is eligible for analysing the dataset in the statistical way.</td></tr>
<tr><td><b>Success Guarantee</b></td><td colspan="2">▪ IF the validation passes without any error.
▪ The selected value should be in the range to display.
▪ A valid output should be displayed.</td></tr>
<tr><td rowspan="4"><b>Flow of events</b></td><td><b>Actors</b></td><td><b>System</b></td></tr>
<tr><td>Import the data set</td><td>Display the dataset</td></tr>
<tr><td>Select the columns which needed to be plotted in the chart</td><td>Verify whether the appropriate column is selected</td></tr>
<tr><td>Display the doughnut chart</td><td></td></tr>
<tr><td><b>Exception condition</b></td><td colspan="2">Unable to select the dataset.
Selecting the inappropriate columns for the line chart
Dynamic allocation of the X axis and Y axis values does not happen
If there is no data in the selected range</td></tr>
</table>

5. ACTIVITY DIAGRAM

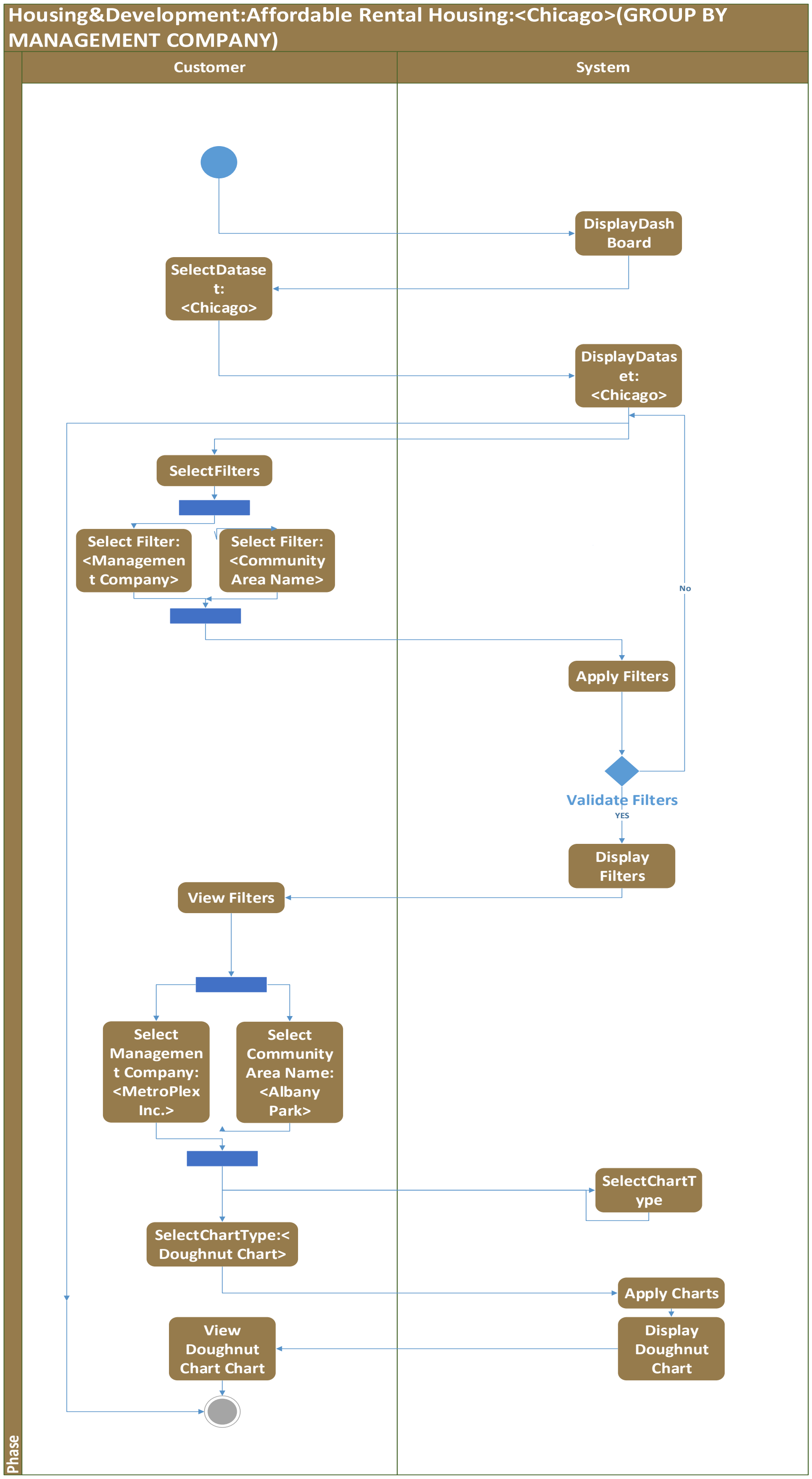
5.1. Affordable Housing Inventory – Austin Dataset


Housing&Development:Affordable Housing Inventory<Austin>
Customer
System
Phase
DisplayDashBoard
SelectDataset:<Austin>
DisplayDataset:<Austin>
SelectFilters
Select Filter:<Developer>
Select Filter:<Total Affordable Units>
Apply Filters
No
Validate Filters
YES
Display Filters
View Filters
Select Developer:<LifeWorks>
Select TotalAffordableUnits:<greater than 50>
SelectChartType
SelectChartType:<Linechart>
Apply Charts
Display Line Chart
View Line Chart

The activity diagram shown above depicts the Affordable Housing Inventory dataset pertaining to Austin city. The Dataset is loaded from the dashboard to display it as a chart.

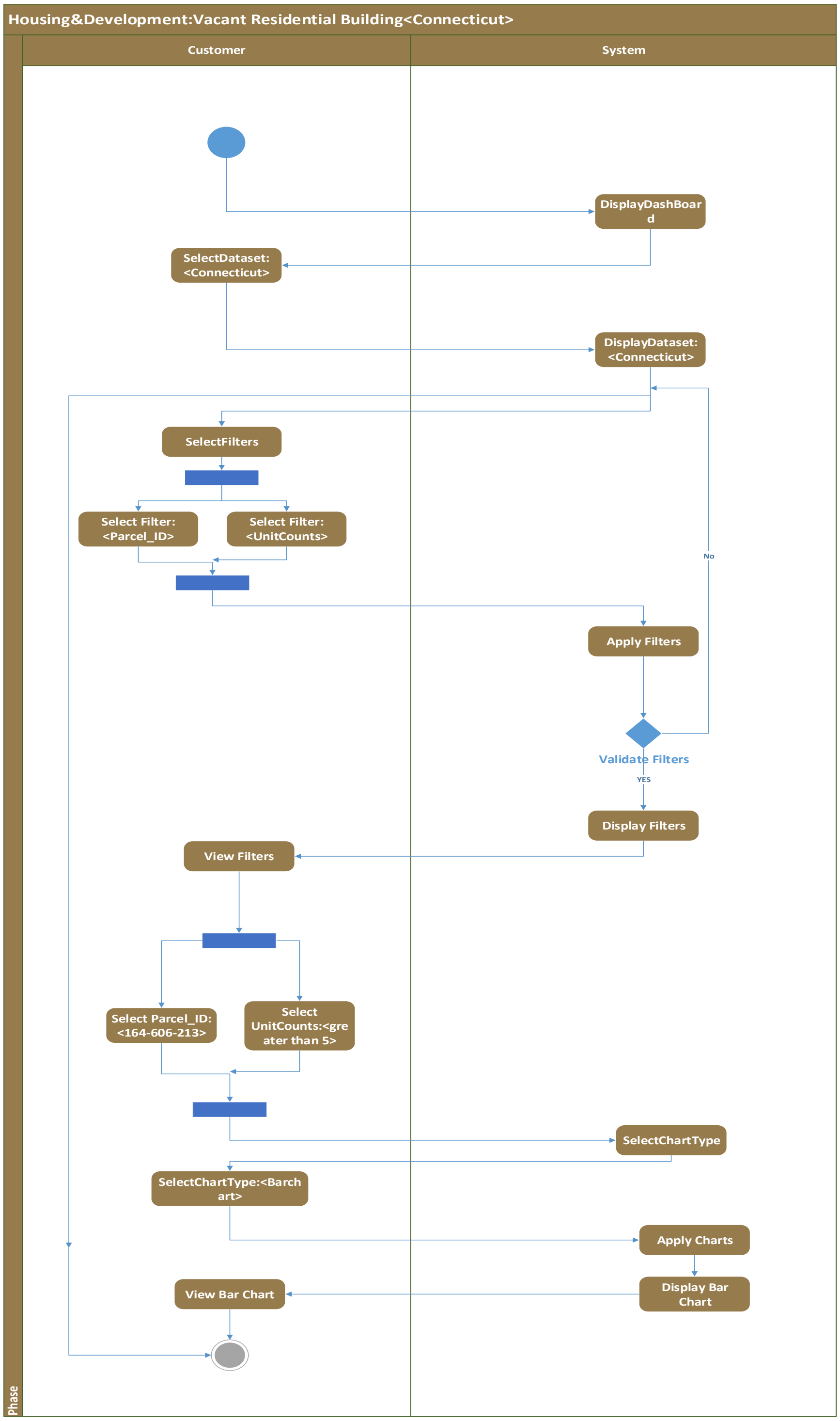
User opens the dashboard selects the dataset Austin from the drop down menu. The dashboard with Austin dataset i.e. Affordable Housing Inventory is displayed. User has the option to slice rows with the help of checkbox on the bottom part of the dashboard and type in the criteria for data selection(Column dicing) at the same time. In our use case we are slicing developer and total affordable units alone. Once the specific values for filtering is selected user can go for chart generation by selecting the type of chart(in our case: Line Chart) and click on Apply Filters. The Filtered/Sliced charts is then displayed.

5.2. Affordable Rental Housing – Chicago Dataset – Selection of chart


Housing&Development:Affordable Rental Housing:<Chicago>(GROUP BY MANAGEMENT COMPANY)
Customer
System
Phase
DisplayDash Board
SelectDataset: <Chicago>
DisplayDataset: <Chicago>
SelectFilters
Select Filter: <Management Company>
Select Filter: <Community Area Name>
No
Apply Filters
Validate Filters
YES
Display Filters
View Filters
Select Management Company: <MetroPlex Inc.>
Select Community Area Name: <Albany Park>
SelectChartType
SelectChartType:< Doughnut Chart>
Apply Charts
Display Doughnut Chart
View Doughnut Chart Chart

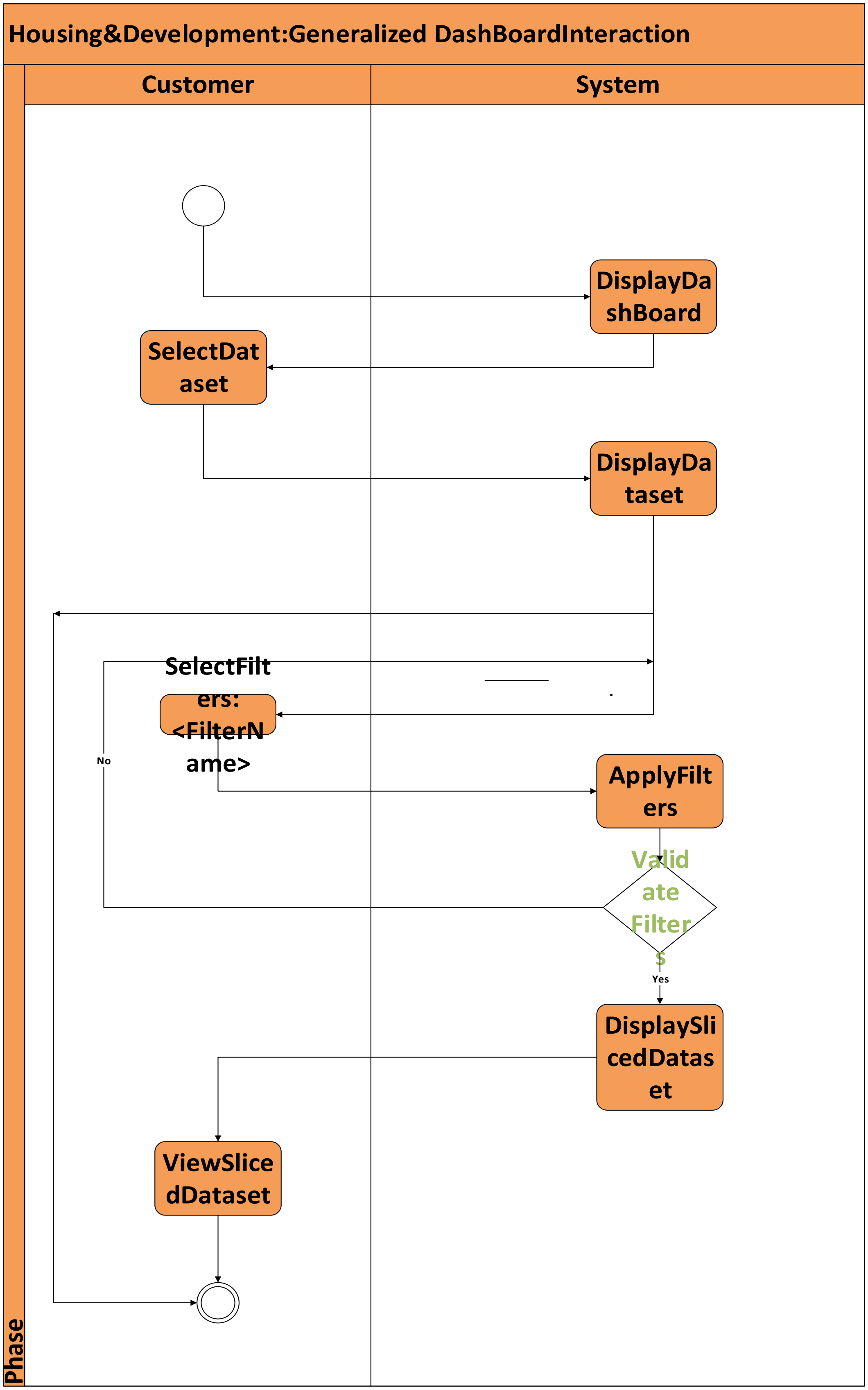
The activity diagram shown above depicts the Affordable Rental Housing dataset pertaining to Chicago city. The Dataset is loaded from the dashboard to display it as a chart.
The user can select the Management company MetroPlex Inc as criteria to filter the data.

5.3. Vacant Residential Building – To plot the chart - Connecticut

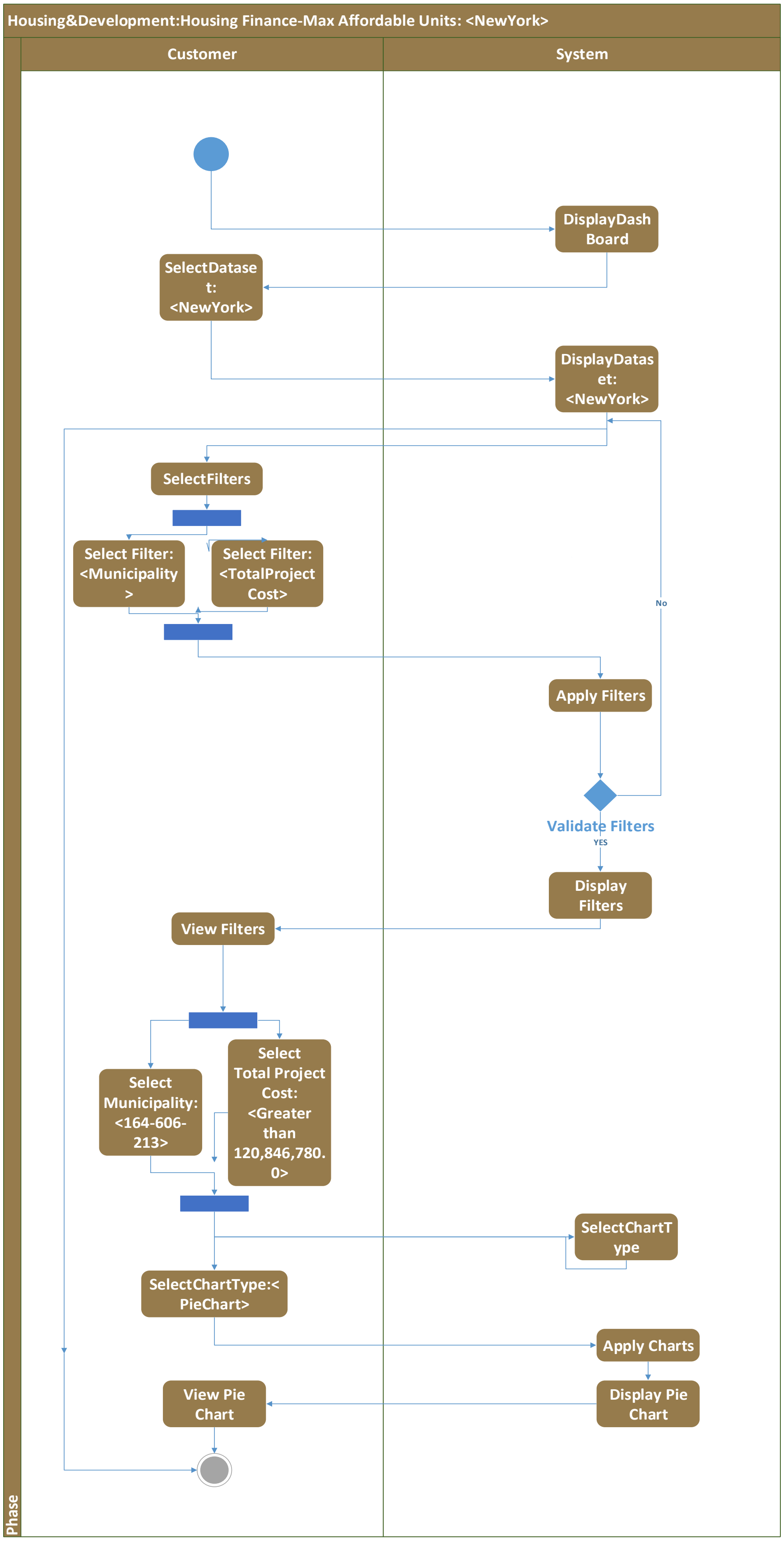

Housing&Development:Vacant Residential Building<Connecticut>
Customer
System
Phase
DisplayDashBoard
SelectDataset: <Connecticut>
DisplayDataset: <Connecticut>
SelectFilters
Select Filter: <Parcel_ID>
Select Filter: <UnitCounts>
No
Apply Filters
Validate Filters
YES
Display Filters
View Filters
Select Parcel_ID: <164-606-213>
Select UnitCounts:<greater than 5>
SelectChartType
SelectChartType:<Barchart>
Apply Charts
Display Bar Chart
View Bar Chart

User opens the dashboard selects the dataset Connecticut from the drop down menu. The dashboard with Connecticut dataset i.e. vacant residential building is displayed. User has the option to slice rows with the help of checkbox on the bottom part of the dashboard and type in the criteria for data selection(Column dicing) at the same time. In here, we are slicing Parcel_ID and units count alone. Once the specific values for filtering is selected user can go for chart generation by selecting the type of chart(in our case: Bar Chart) and click on Apply Filters. The Filtered/Sliced charts is then displayed.

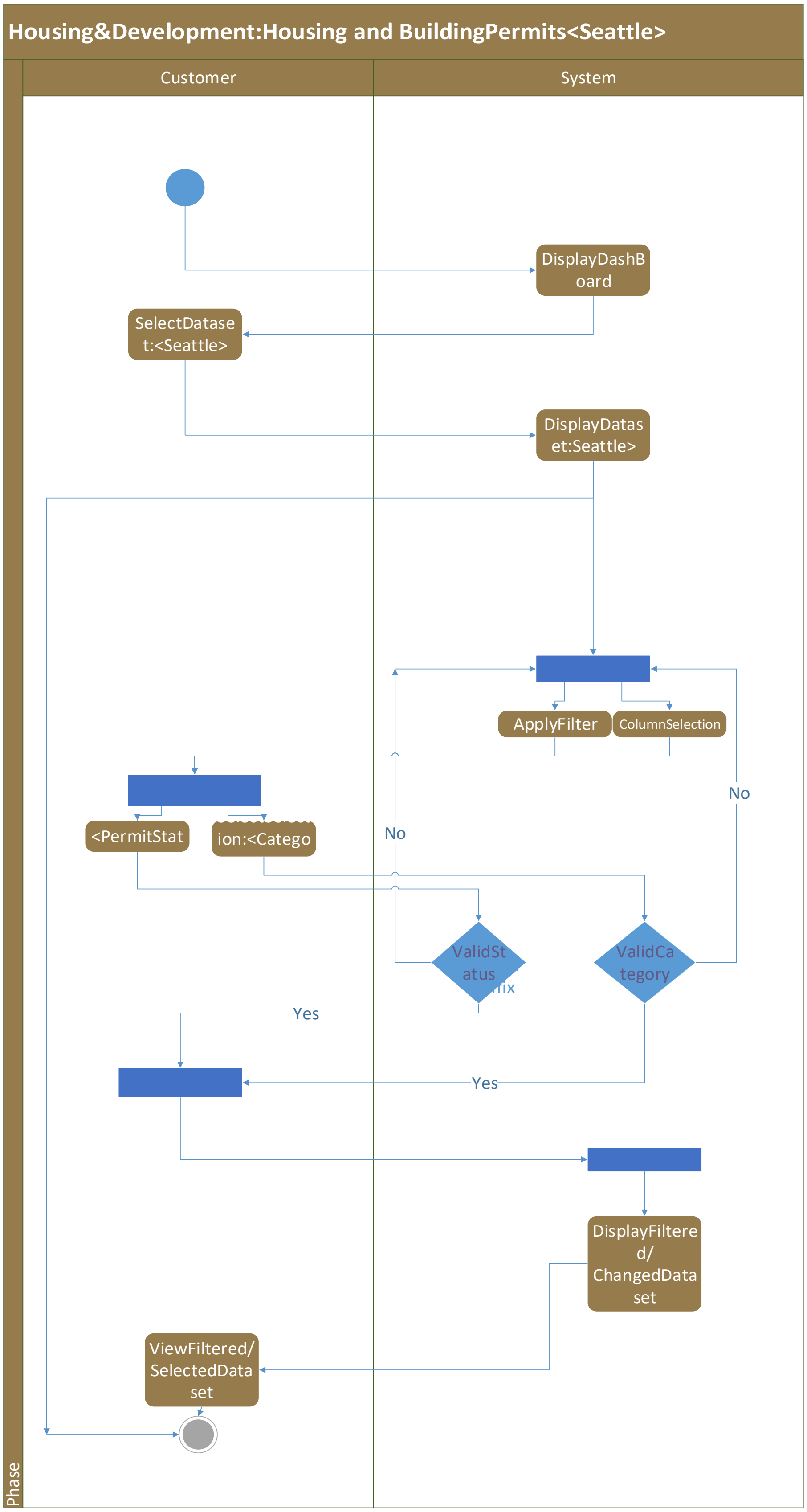
5.4. Housing & Development Dashboard – General Flow



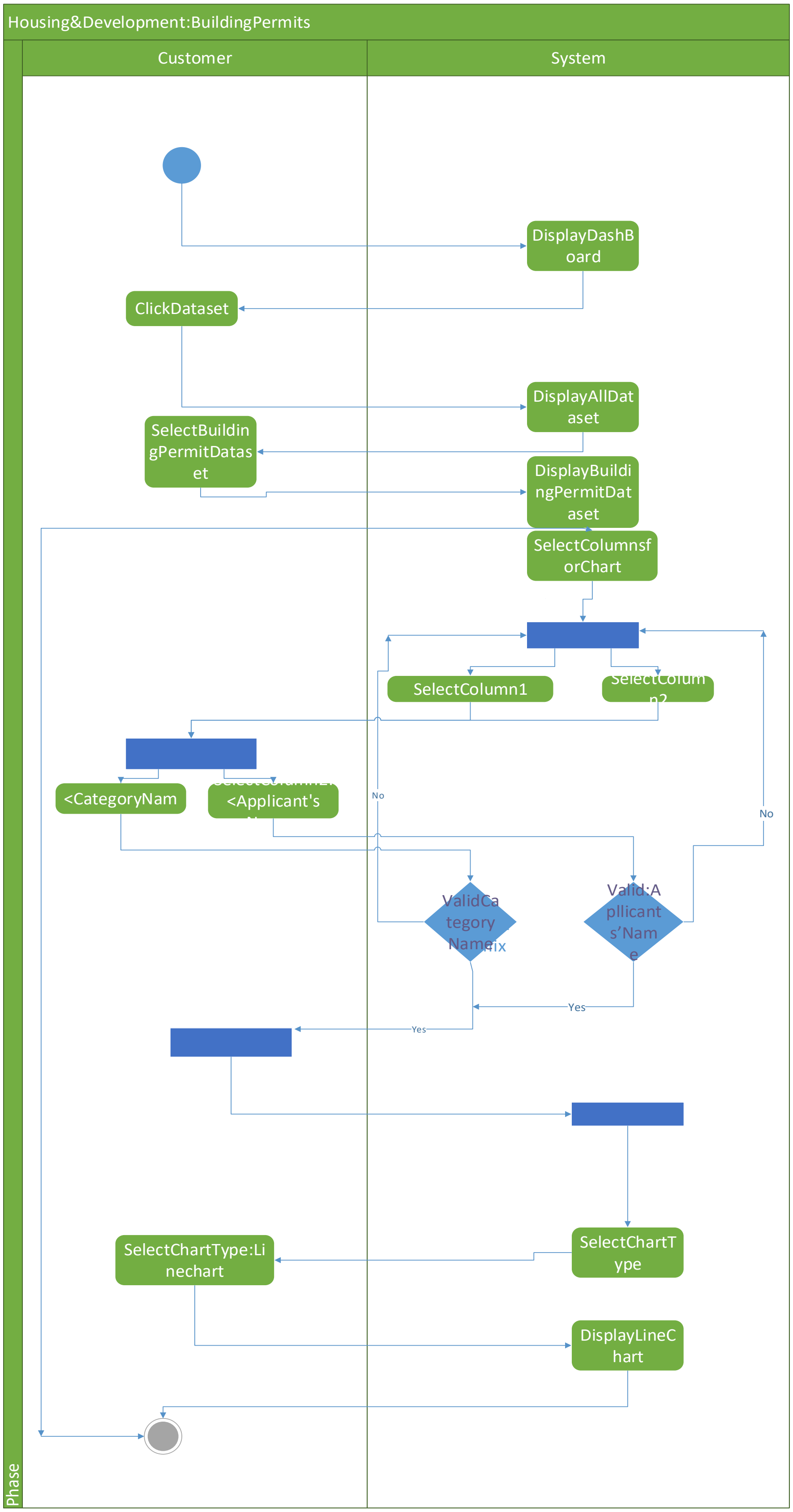
5.5. Housing Finance – Max Affordable units – New York


Housing&Development:Housing Finance-Max Affordable Units: <NewYork>
Customer
System
DisplayDash Board
SelectDataset: <NewYork>
DisplayDataset: <NewYork>
SelectFilters
Select Filter: <Municipality >
Select Filter: <TotalProject Cost>
No
Apply Filters
Validate Filters
YES
Display Filters
View Filters
Select Municipality: <164-606-213>
Select Total Project Cost: <Greater than 120,846,780.0>
SelectChartType
SelectChartType:<PieChart>
Apply Charts
View Pie Chart
Display Pie Chart
Phase

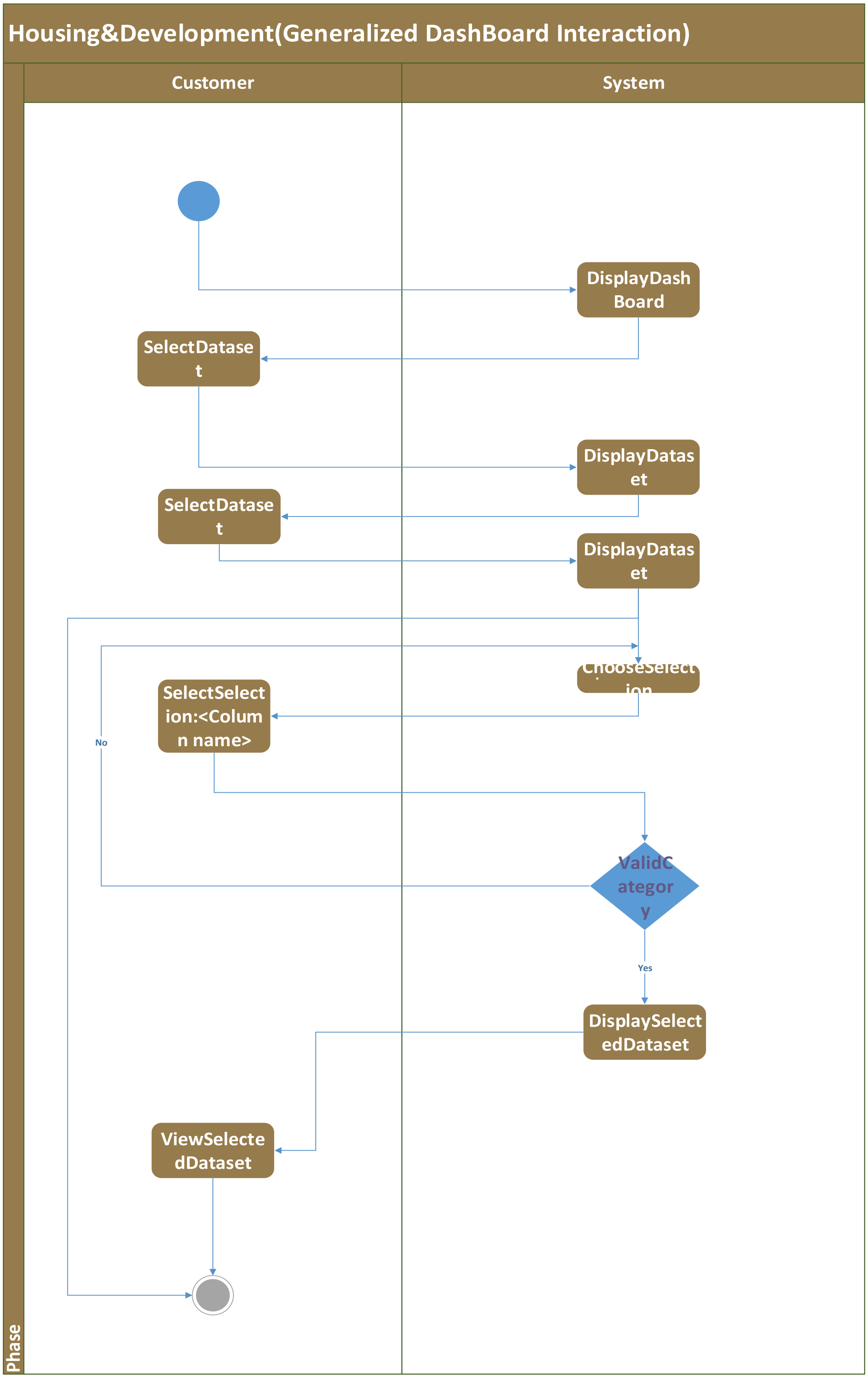
5.6. Housing and Development – Building Permits – apply filter- Seattle



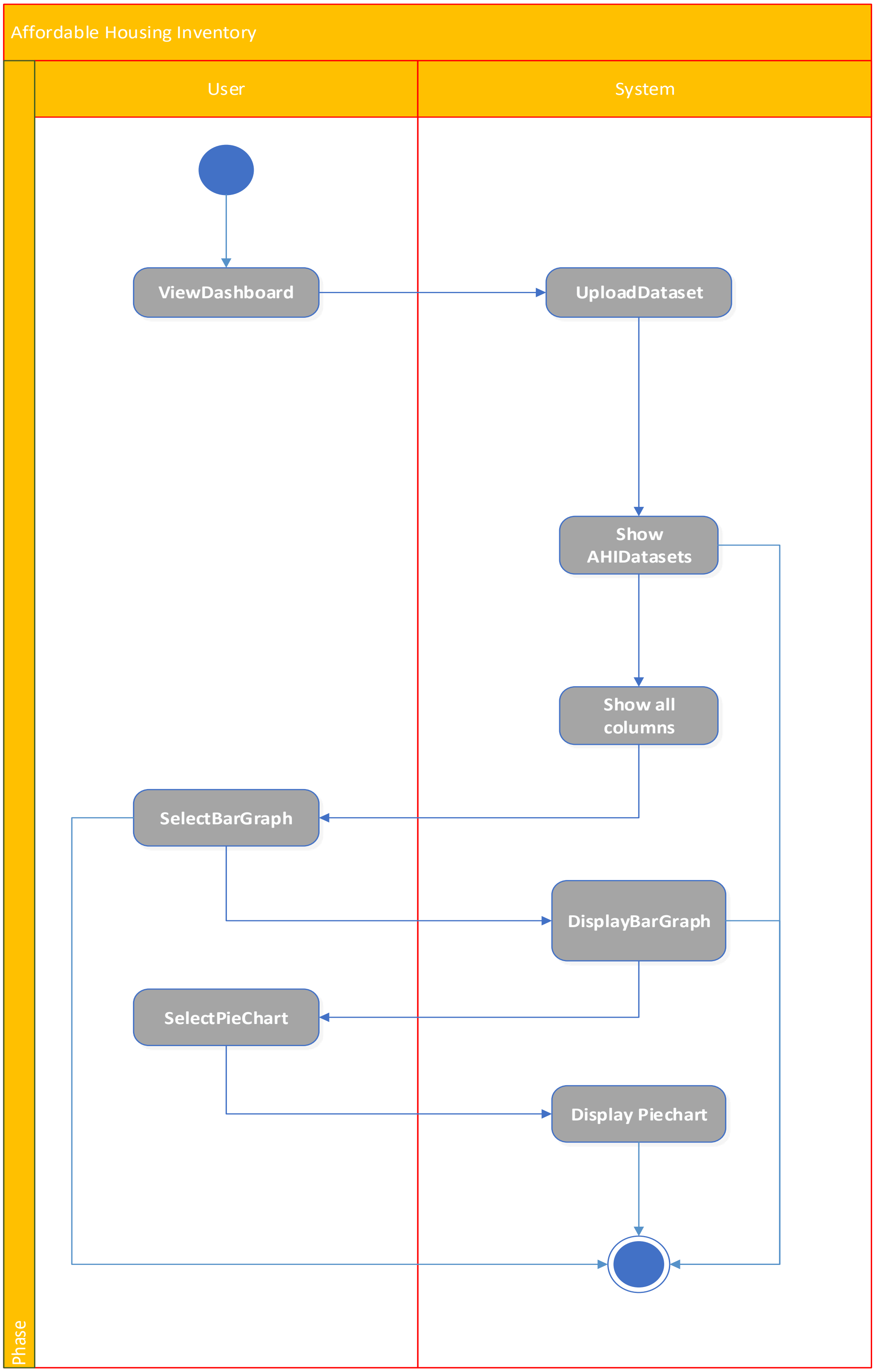
5.7. Housing and Development – Building Permits – Plot Chart



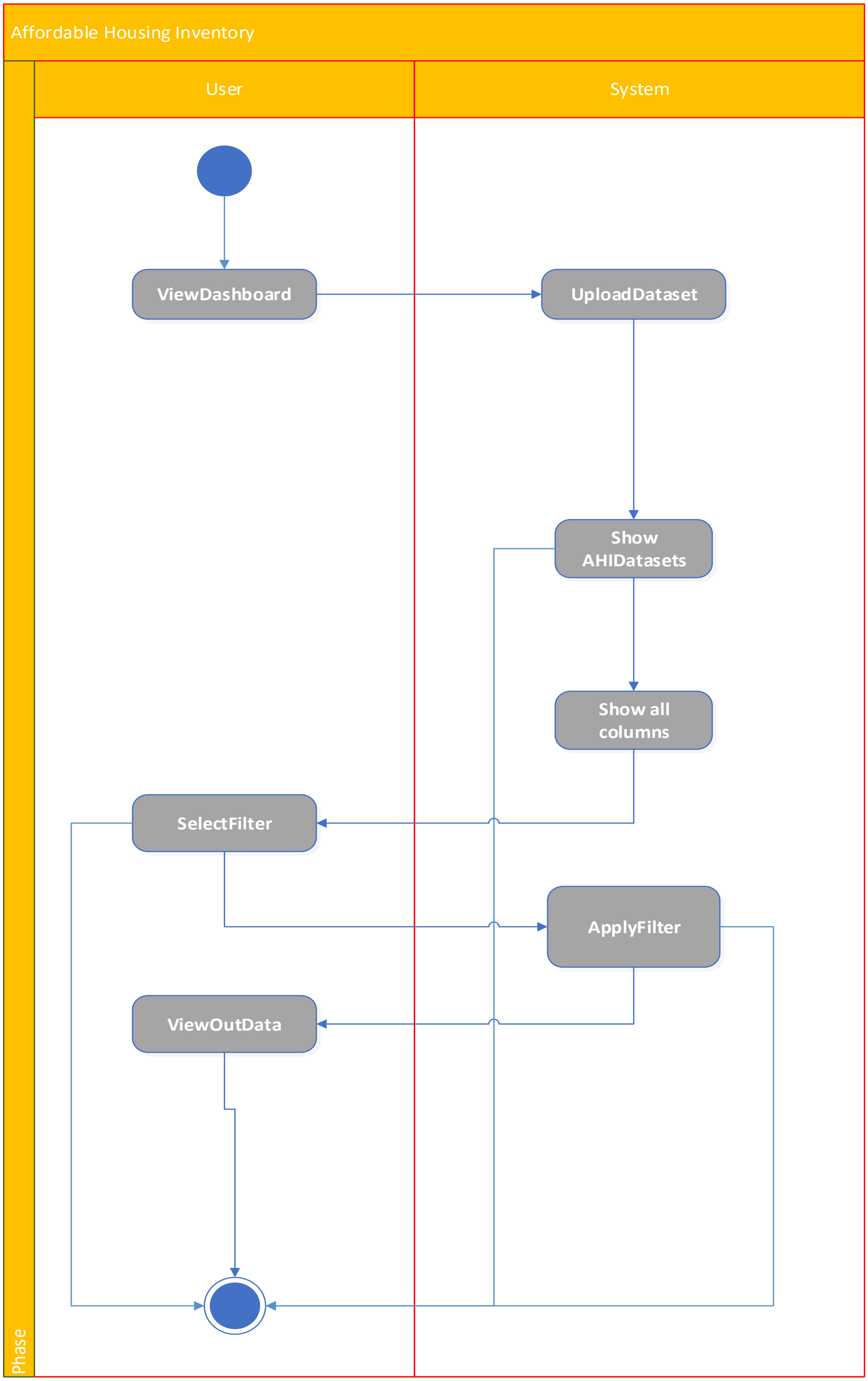
5.8. Housing & Development – Generalised Dashboard Interaction



5.9. Affordable Housing Inventory – Plot chart

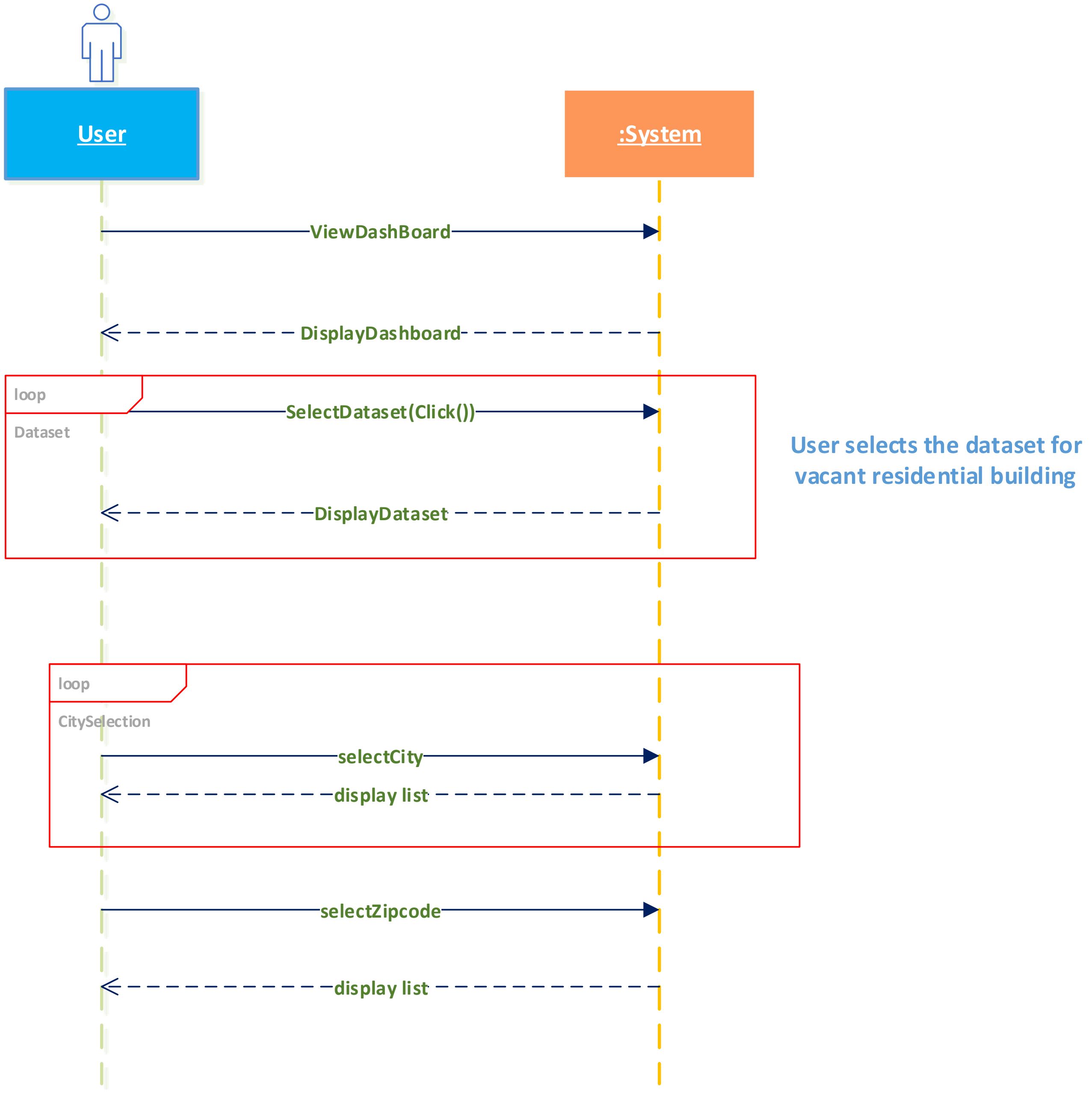


5.10. Affordable Housing Inventory – Apply filter

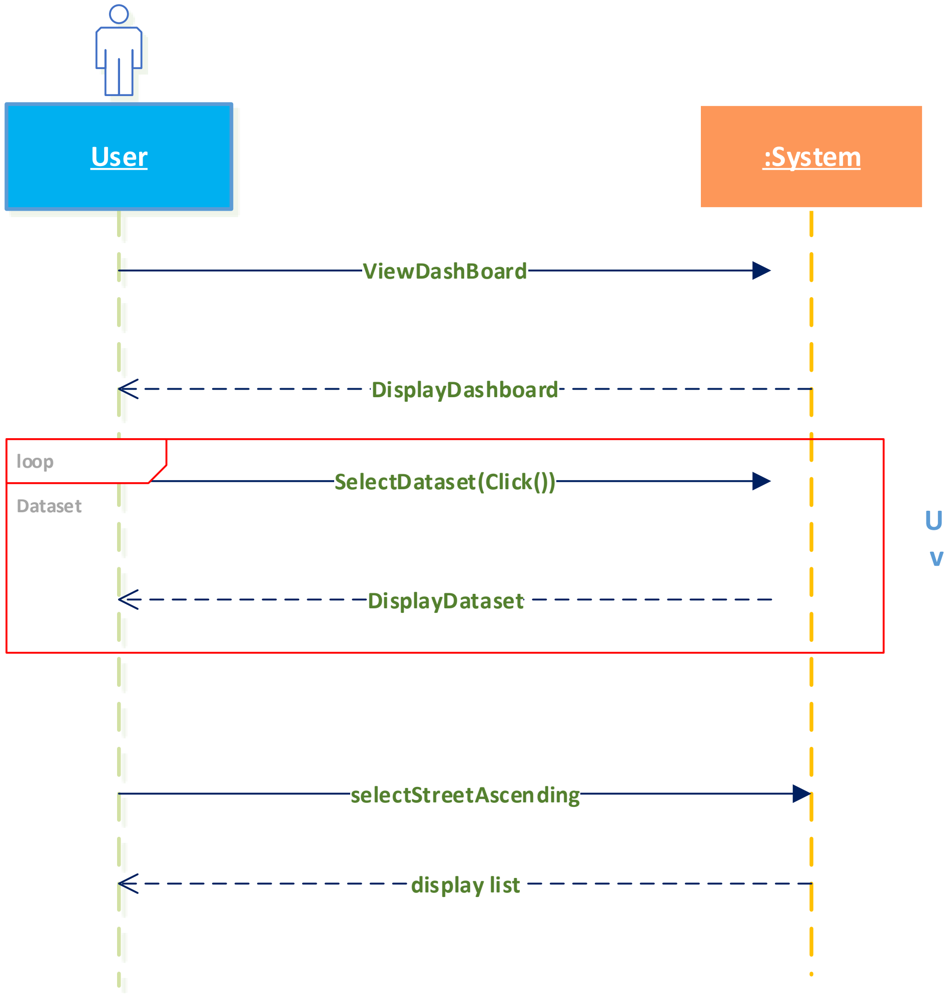


6. SYSTEM SEQUENCE DIAGRAM:

6.1 . VACANT RESIDENTIAL BUILDING – CONNECTICUT – SELECTION OF DATA

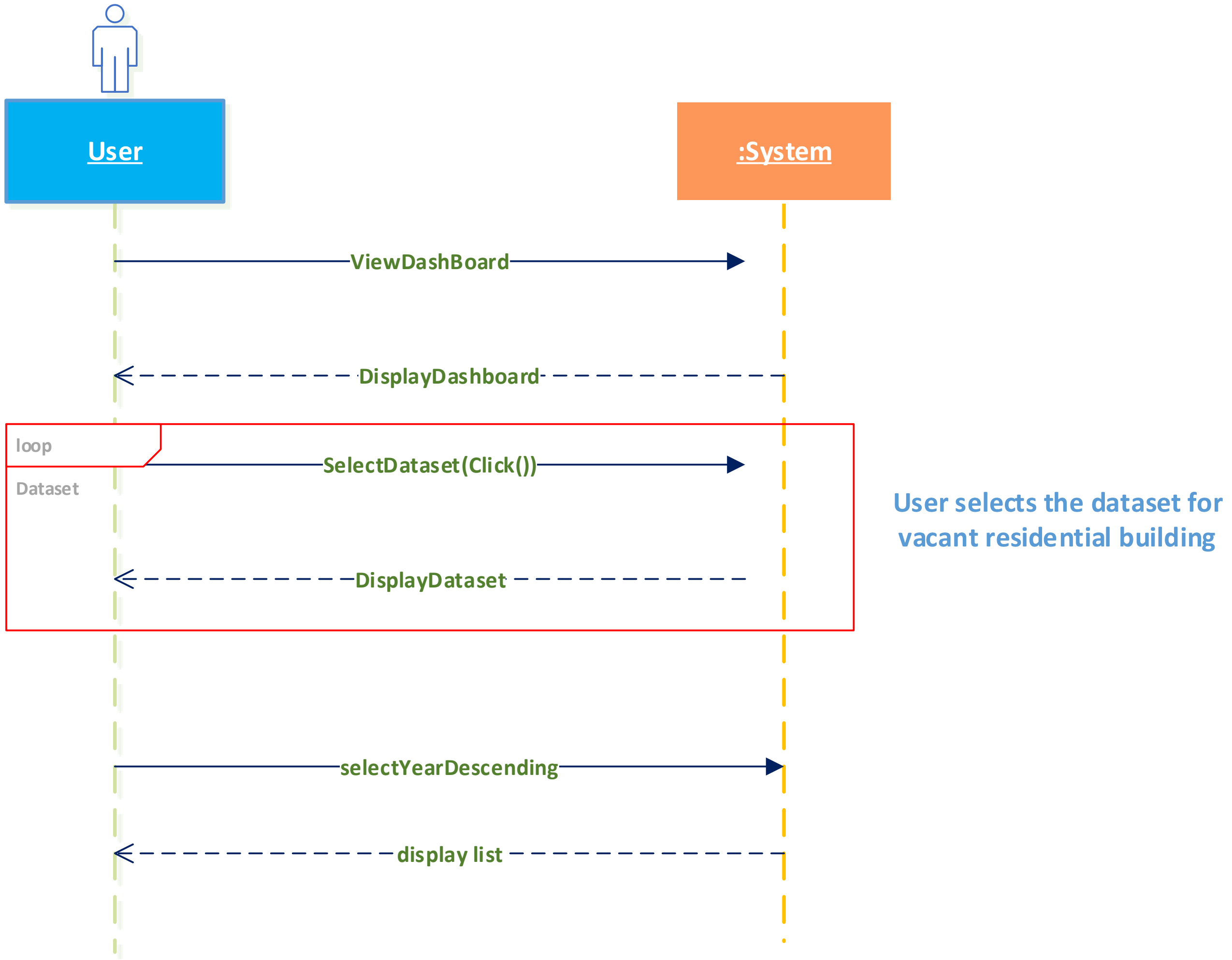


6.2 . VACANT RESIDENTIAL BUILDING – CONNECTICUT – FILTERING OF DATA



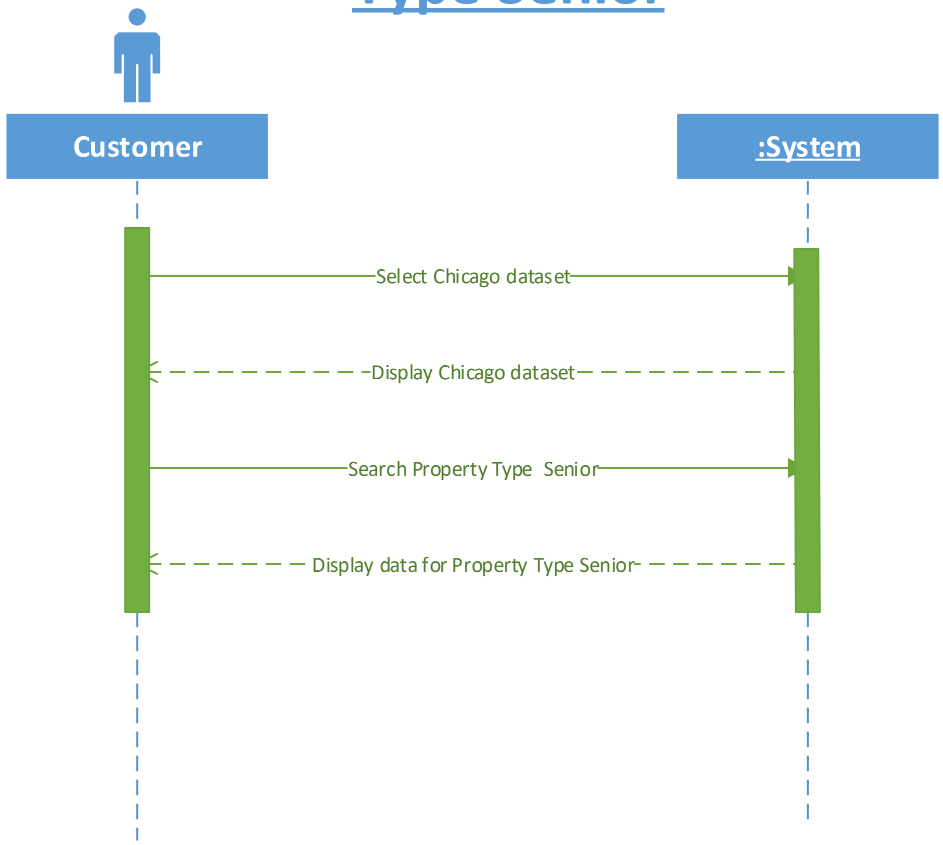
User selects the dataset for vacant residential building

6.3 . VACANT RESIDENTIAL BUILDING – CONNECTICUT – SORTING



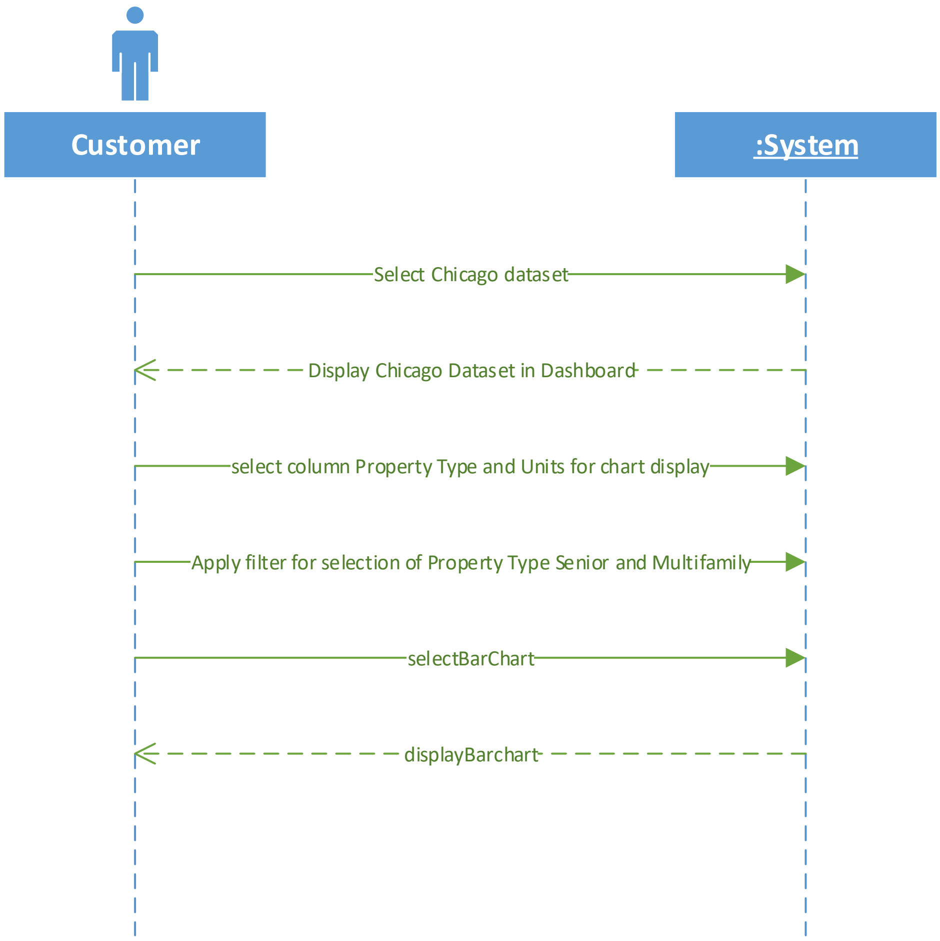
6.4 . RENTAL HOUSING – CHICAGO -PROPERTY TYPE SENIOR

Search For Property Type Senior

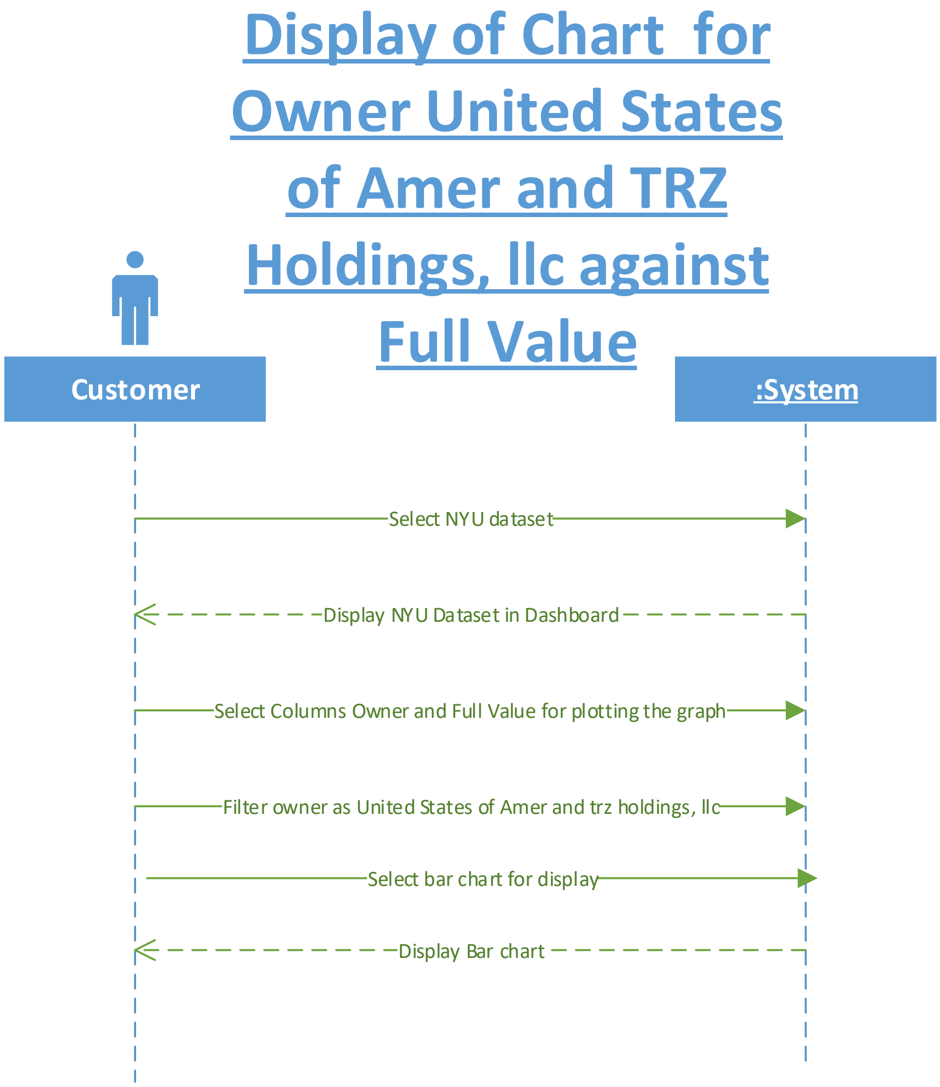

Customer
:System
Select Chicago dataset
Display Chicago dataset
Search Property Type Senior
Display data for Property Type Senior

6.5 . RENTAL HOUSING – CHICAGO -PLOTTING CHART

Plotting Chart

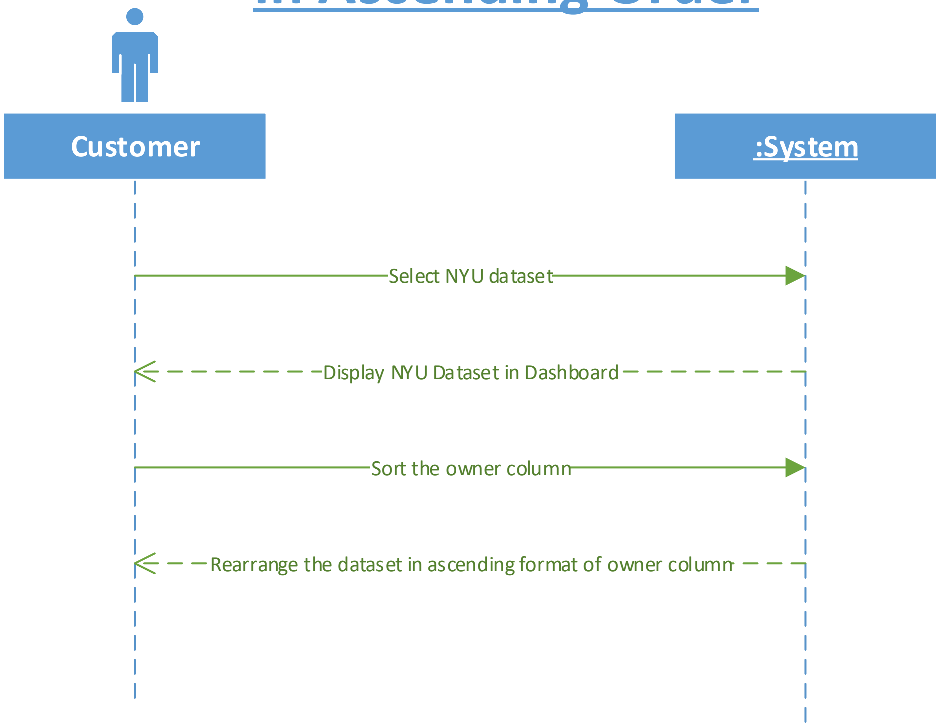

Customer
:System
Select Chicago dataset
Display Chicago Dataset in Dashboard
select column Property Type and Units for chart display
Apply filter for selection of Property Type Senior and Multifamily
selectBarChart
displayBarchart

6.6 . NEW YORK – HOUSING FINANCE –


Display of Chart for Owner United States of Amer and TRZ Holdings, llc against Full Value
Customer
:System
Select NYU dataset
Display NYU Dataset in Dashboard
Select Columns Owner and Full Value for plotting the graph
Filter owner as United States of Amer and trz holdings, llc
Select bar chart for display
Display Bar chart

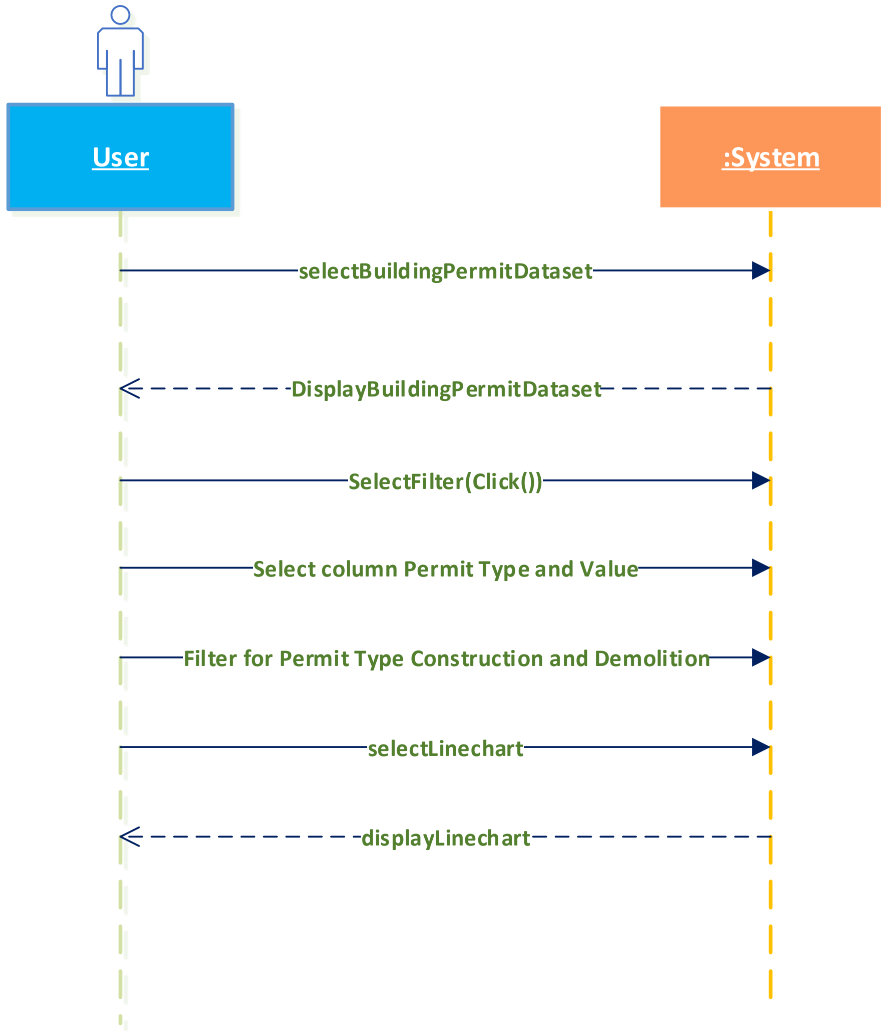
6.7 . NEW YORK – HOUSING FINANCE –

Display Owner Name in Ascending Order


Customer
:System
Select NYU dataset
Display NYU Dataset in Dashboard
Sort the owner column
Rearrange the dataset in ascending format of owner column

6.8 . BUILDING PERMIT – SEATTLE - PLOT LINE CHART

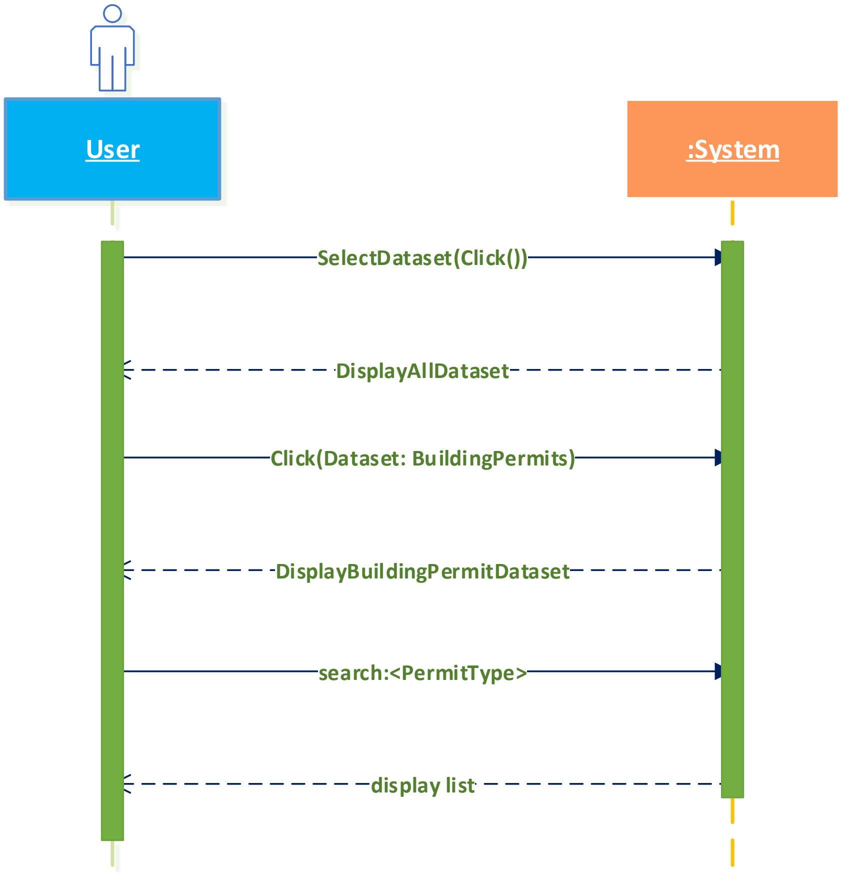
Assuming the Dataset Lists is listed on DashBoard


User
:System
selectBuildingPermitDataset
DisplayBuildingPermitDataset
SelectFilter(Click())
Select column Permit Type and Value
Filter for Permit Type Construction and Demolition
selectLinechart
displayLinechart

User selects the dataset for plotting line chart

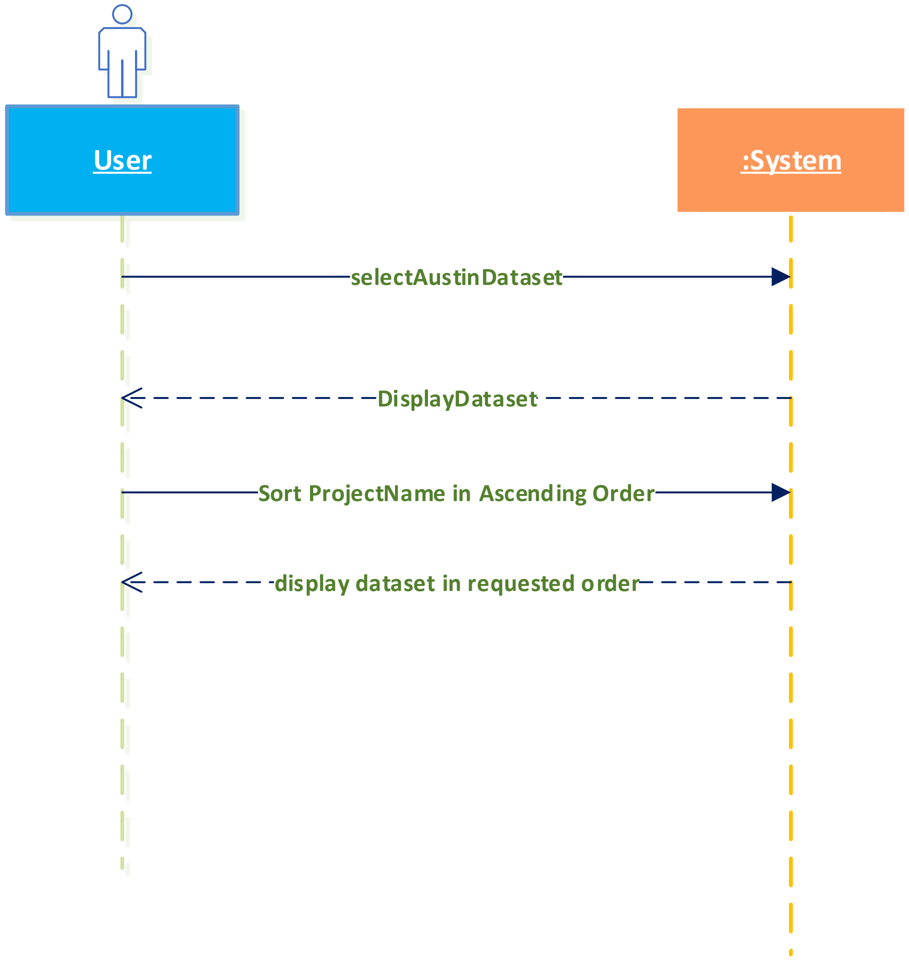
6.9. BUILDING PERMIT – SEATTLE - DISPLAY DATA SET/FILTER

Assuming Main Page is already loaded



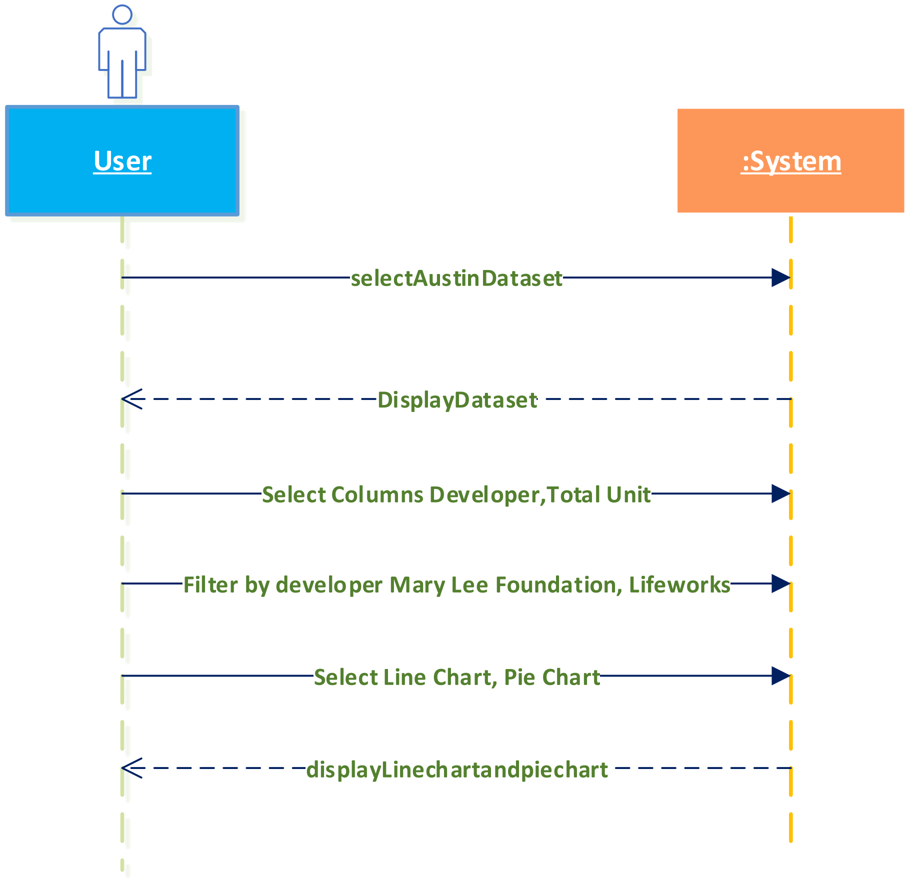
User selects the dataset for PermitType slicing on Building Permits

6.9. AFFORDABLE HOUSING INVENTORY – AUSTIN – FILTER PROJ NAME


User
:System
selectAustinDataset
DisplayDataset
Sort ProjectName in Ascending Order
display dataset in requested order

User sorts ProjectName in Ascending Order

6.10. AFFORDABLE HOUSING INVENTORY – PLOTTING GRAPH


User
:System
selectAustinDataset
DisplayDataset
Select Columns Developer,Total Unit
Filter by developer Mary Lee Foundation, Lifeworks
Select Line Chart, Pie Chart
displayLinechartandpiechart

User selects the dataset for plotting line chart

7. LIST OF CLASSES

SL NO.	CLASS NAME
1	ChartMain
2	ChartFactory
3	ChartOperation
4	ChartConfig
5	ChartDecorator
6	PieChart
7	BarChart
8	LineChart
9	Donoughnut Chart
10	PieChartDecorator
11	BarChartDecorator
12	LineChartDecorator
13	Donoughnut ChartDecorator
14	Austin
15	Chicago
16	Connecticut
17	NewYork
18	Seattle
19	Dataset
20	DataObject

8. CLASS DIAGRAM FOR ANALYSIS MODEL

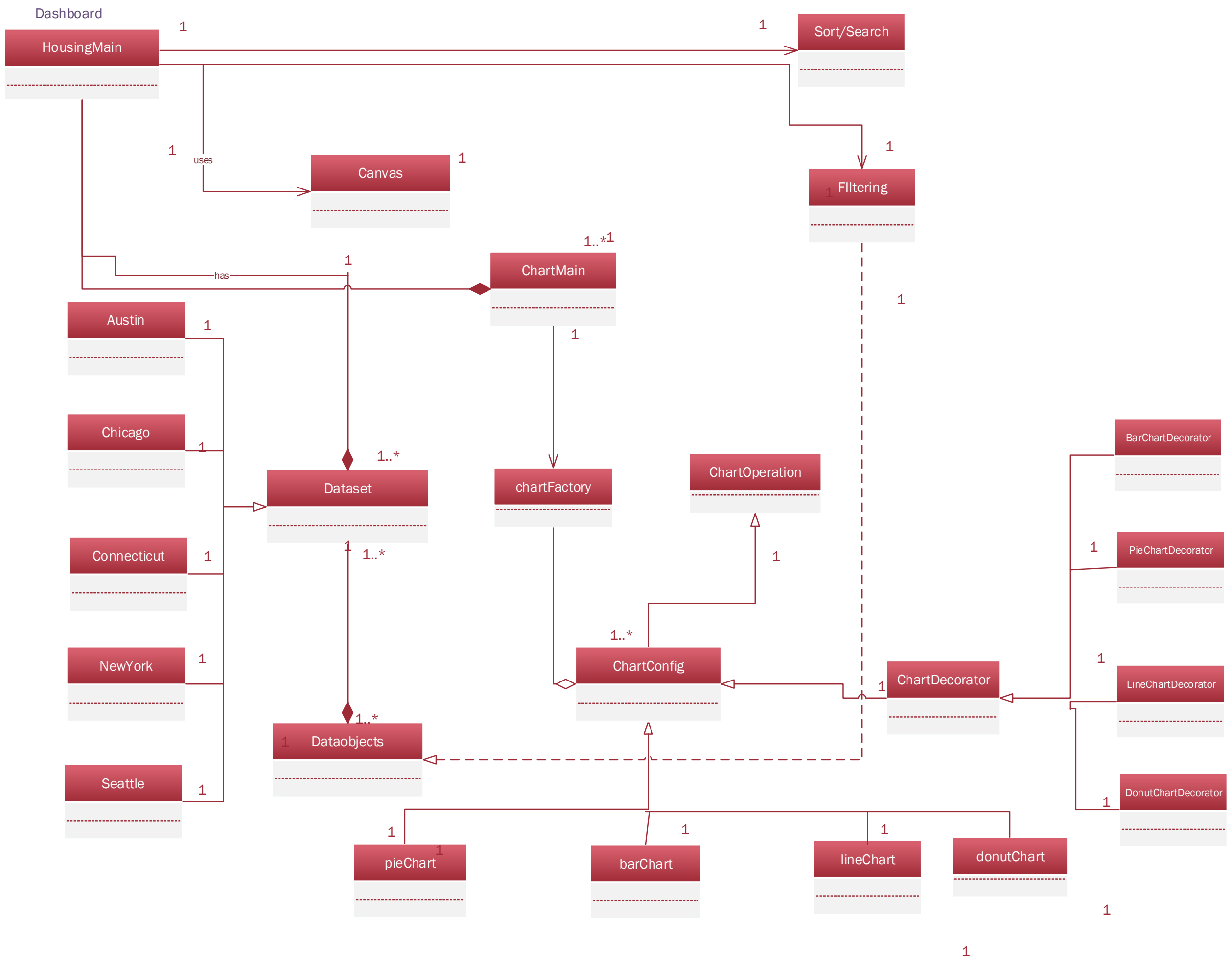


Fig: Class diagram for Analysis model

9. CLASS DIAGRAM FOR DESIGN MODEL

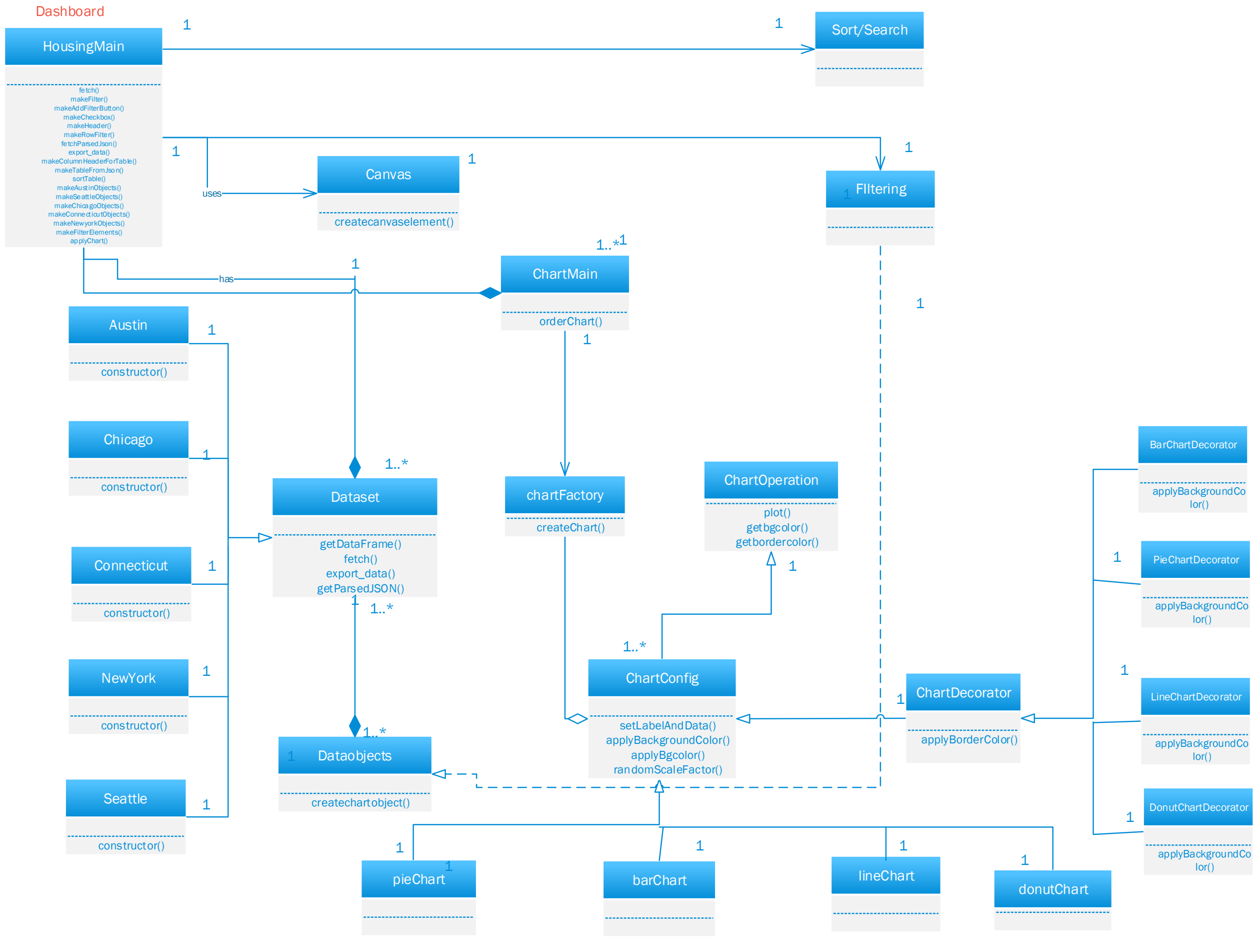


Fig: Class diagram for design model

10. PACKAGE DIAGRAM:

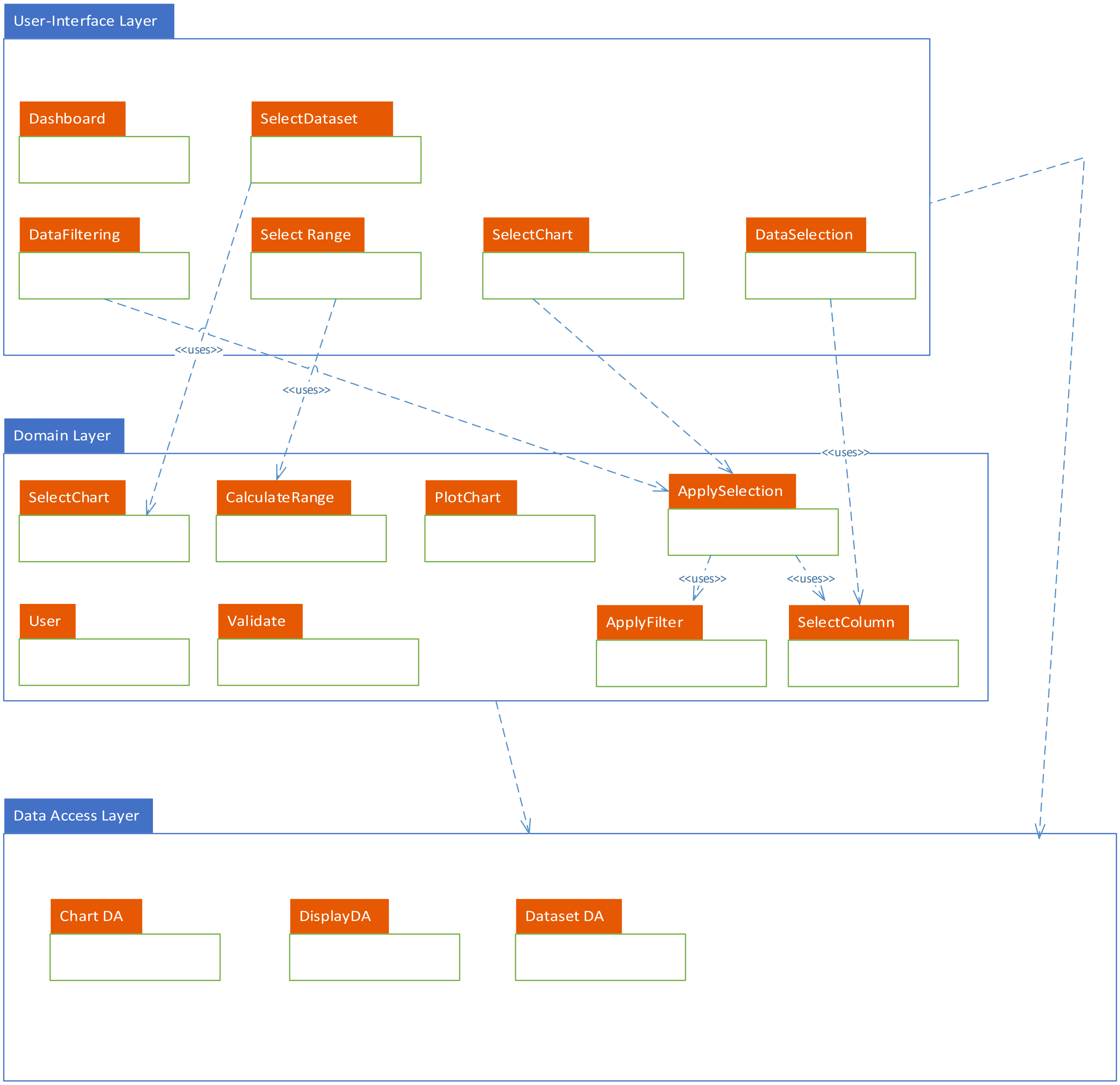
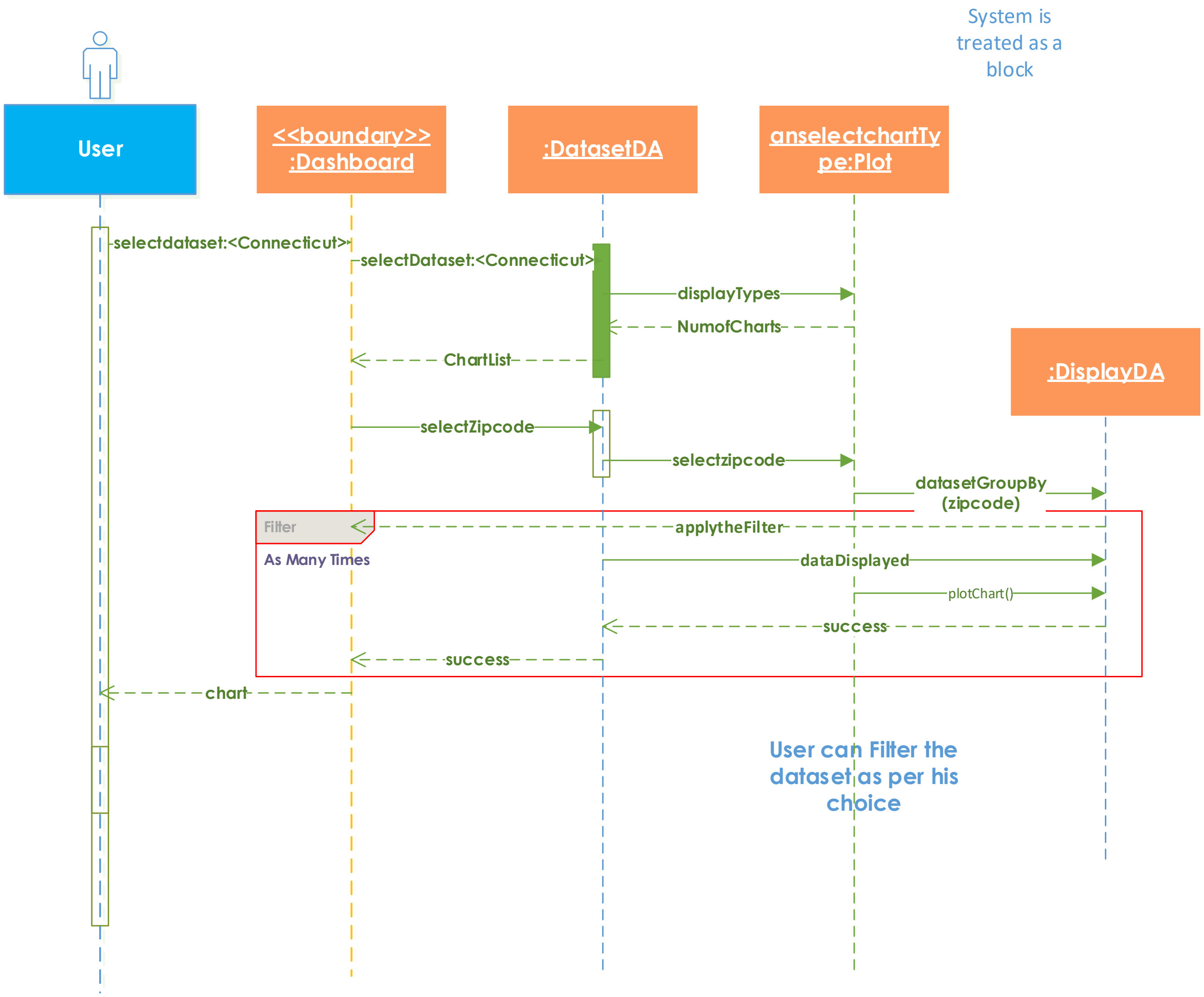


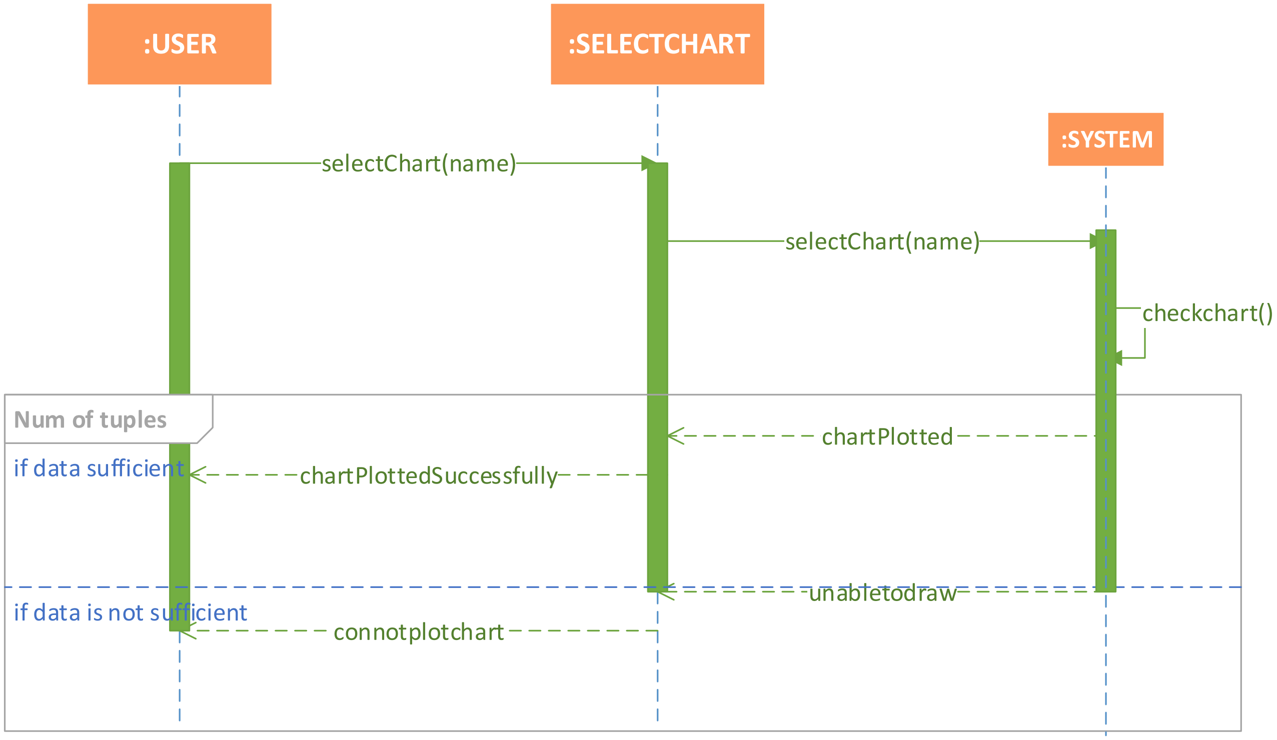
Fig: Package Diagram

11. SEQUENCE INTERACTION DIAGRAM:

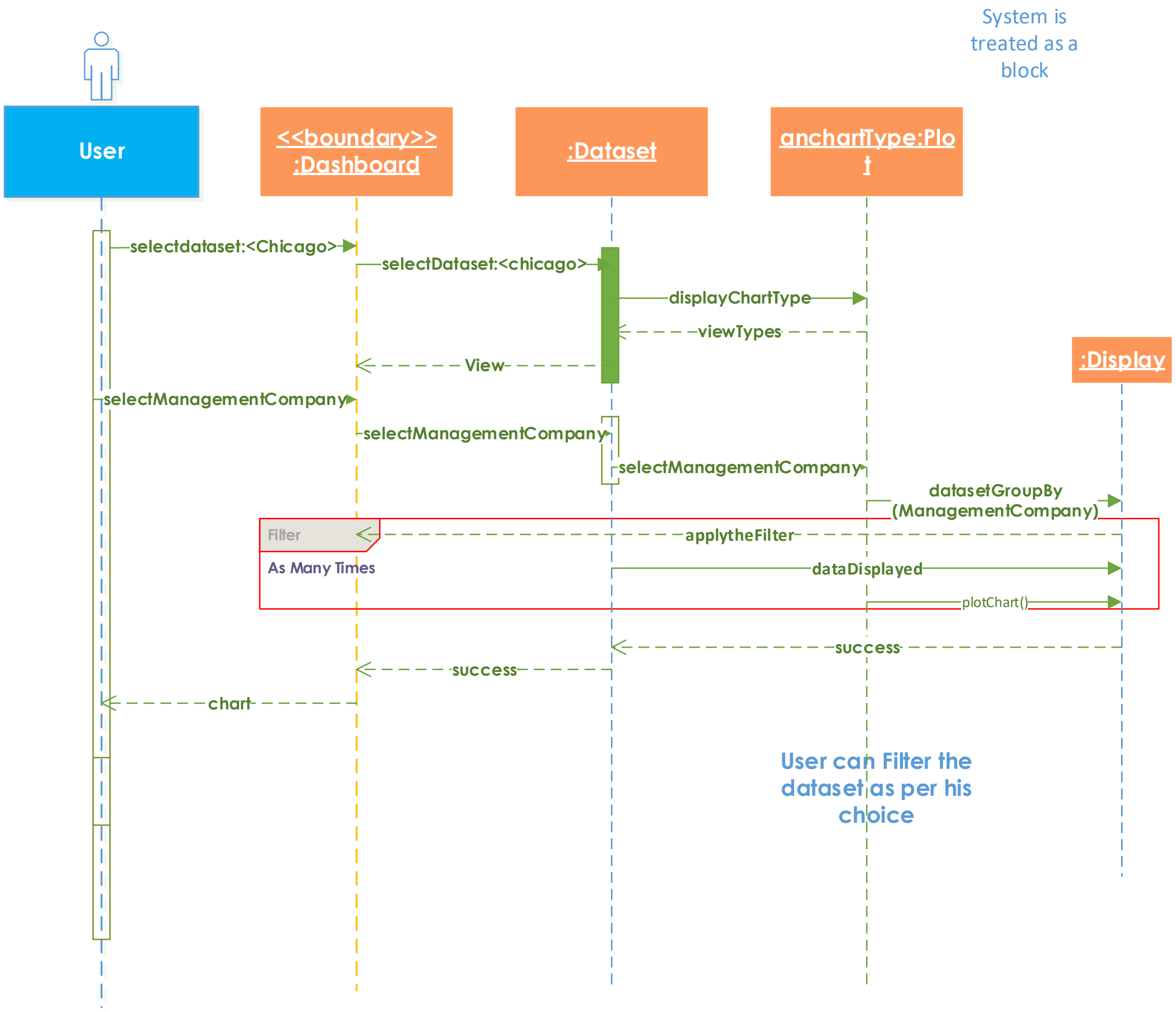
11.1 . VACANT RESIDENTIAL BUILDING – CONNECTICUT – PLOT CHART BASED ON ZIP CODE



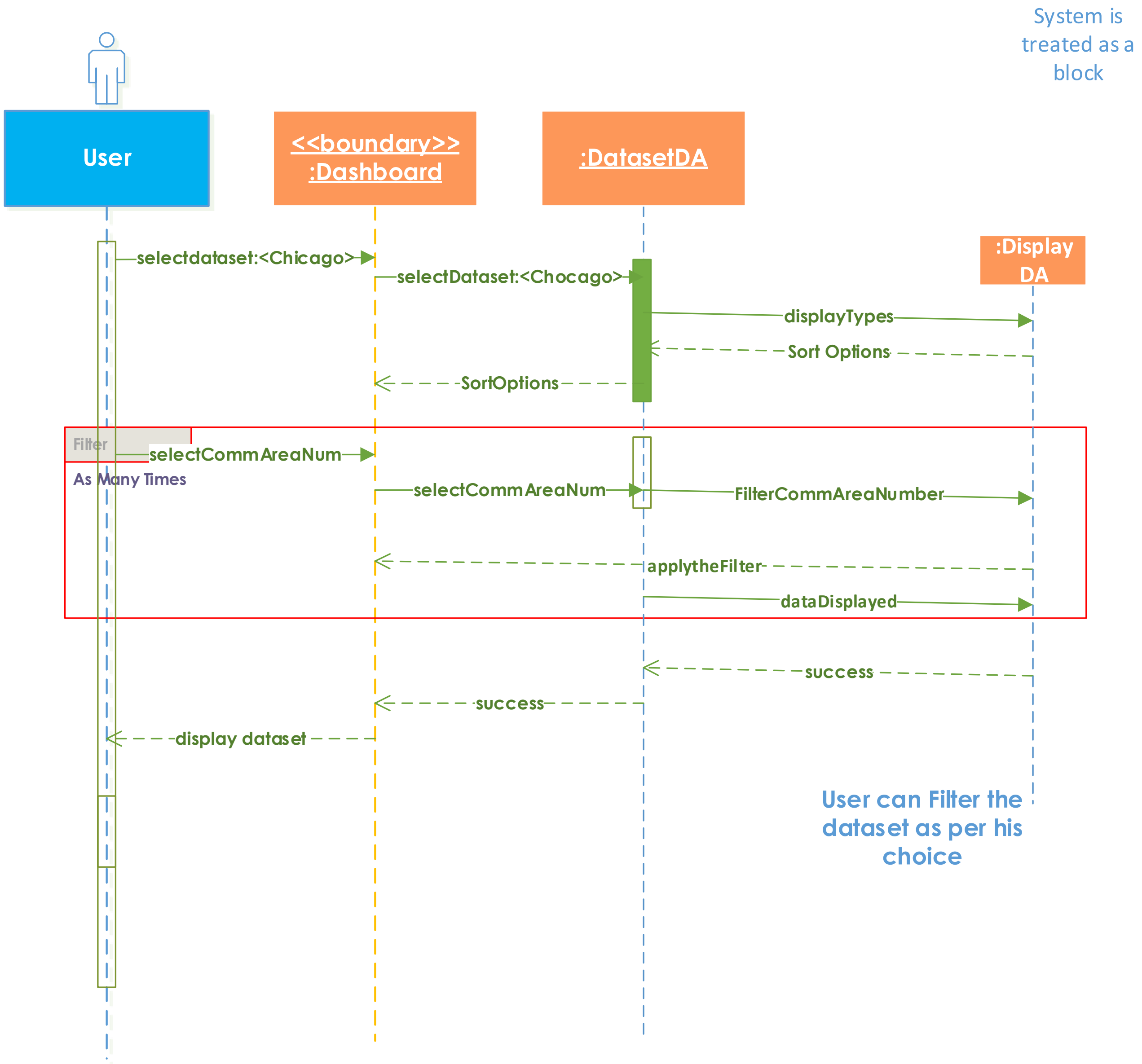
11.2. VACANT RESIDENTIAL BUILDING – CONNECTICUT – PLOTTING OF DATA



11.3. RENTAL HOUSING – CHICAGO -FILTERING OF DATASET

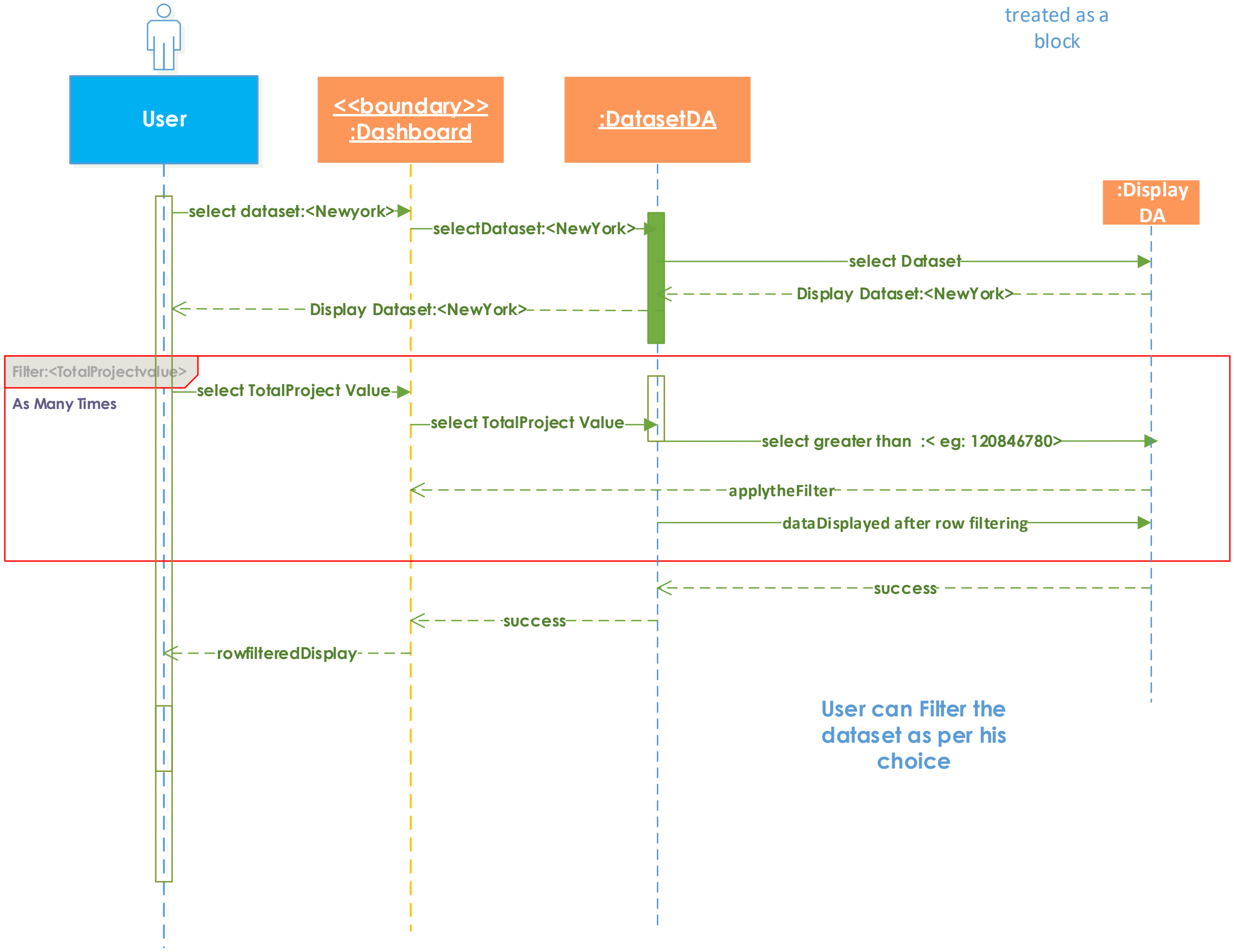


11.4. RENTAL HOUSING – CHICAGO -FILTERING ON COMMONAREANUM



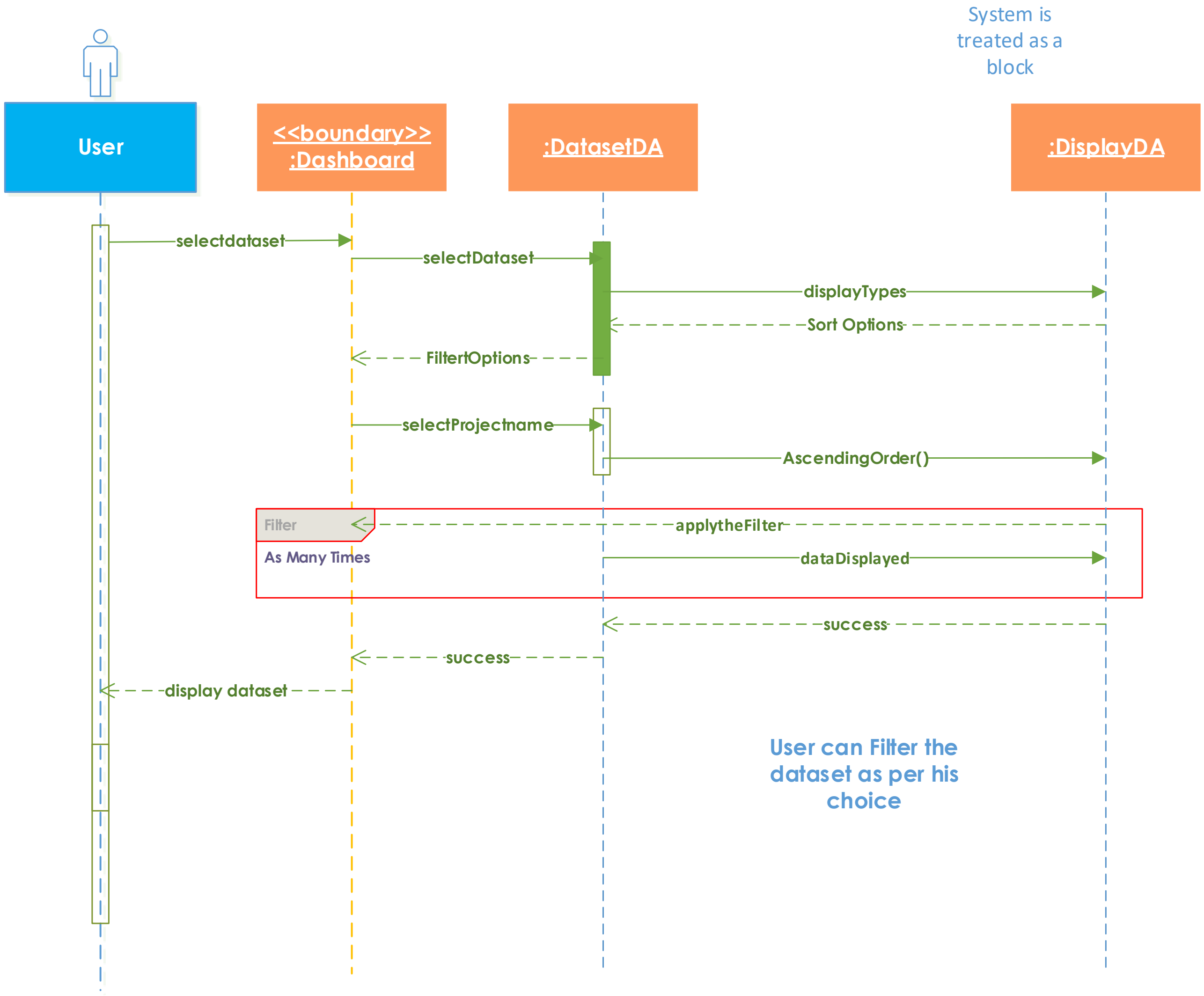
11.5 . NEW YORK – HOUSING FINANCE – ROW FILTERING ON TOTAL PROJ VALUE

System is treated as a block

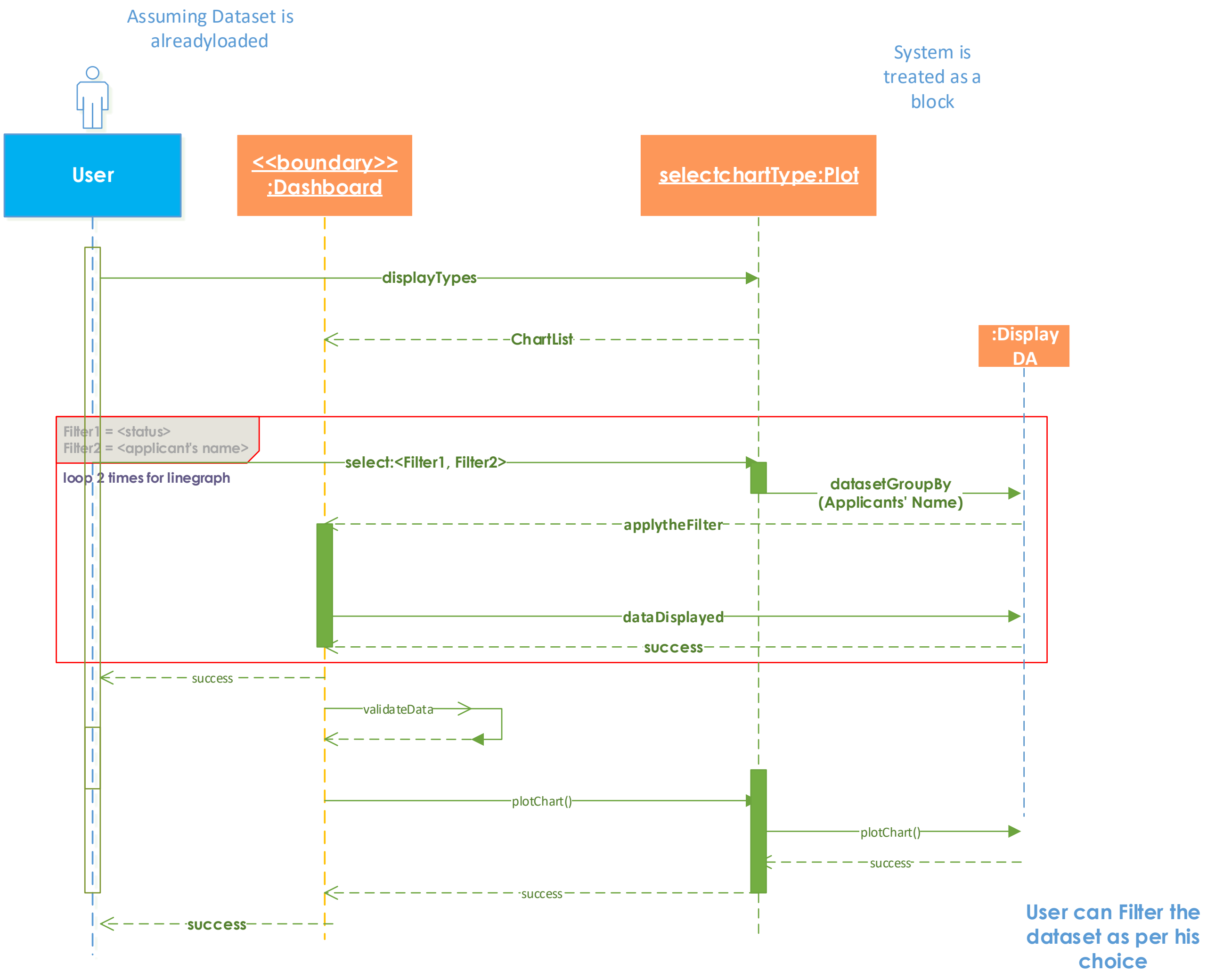

User
<<boundary>> :Dashboard
:DatasetDA
:Display DA
select dataset:<Newyork>
selectDataset:<NewYork>
select Dataset
Display Dataset:<NewYork>
Display Dataset:<NewYork>
Filter:<TotalProjectvalue>
As Many Times
select TotalProject Value
select TotalProject Value
select greater than :< eg: 120846780>
applytheFilter
dataDisplayed after row filtering
success
success
rowfilteredDisplay

User can Filter the dataset as per his choice

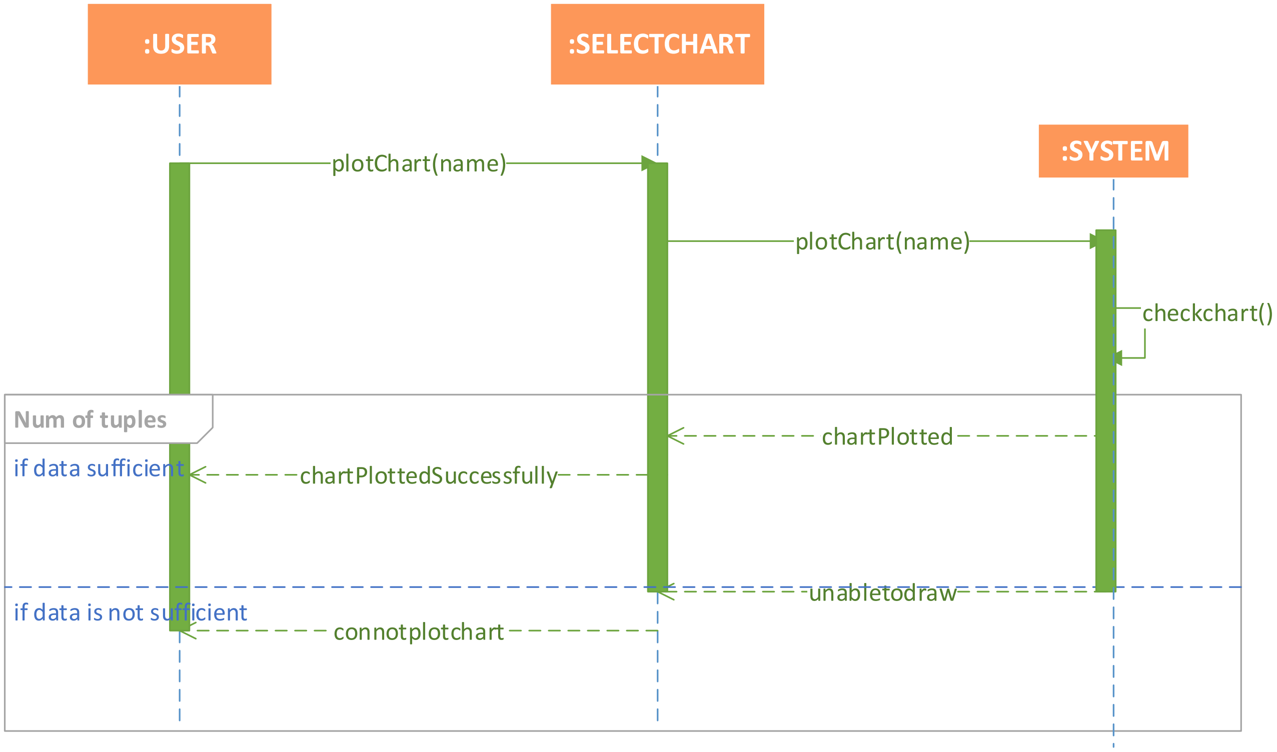
11.6. NEW YORK – HOUSING FINANCE – APPLY FOR FILTER –(ASCENDING ORDERING)



11.7. BUILDING PERMIT – SEATTLE - FILTERING BASED ON APPLICANT'S NAME (GROUP BY)


Assuming Dataset is alreadyloaded
System is treated as a block
User
<<boundary>> :Dashboard
selectchartType:Plot
:Display DA
displayTypes
ChartList
Filter1 = <status>
Filter2 = <applicant's name>
loop 2 times for linegraph
select:<Filter1, Filter2>
datasetGroupBy (Applicants' Name)
applytheFilter
dataDisplayed
success
success
validateData
plotChart()
plotChart()
success
success
success
User can Filter the dataset as per his choice

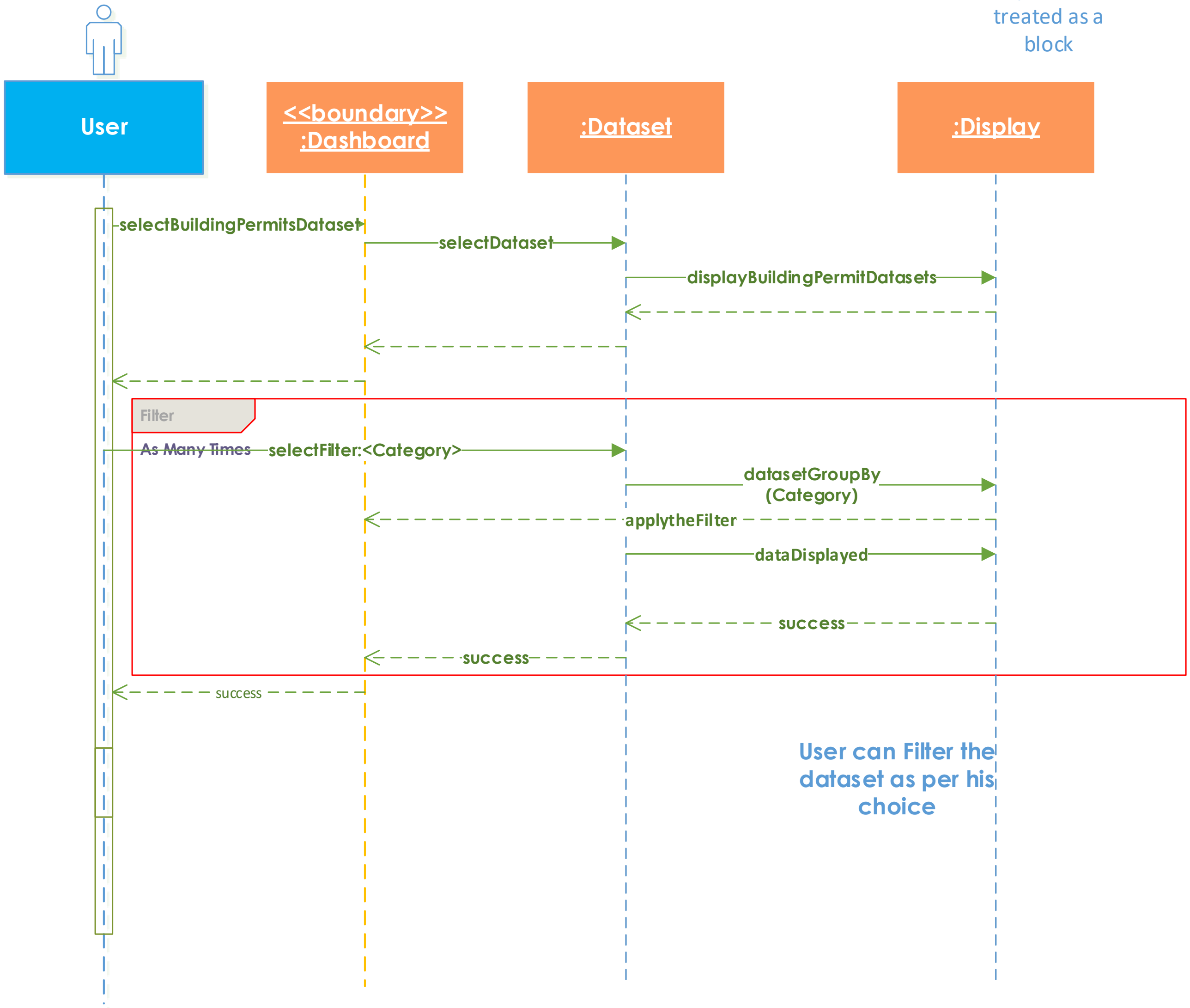
11.8. BUILDING PERMIT – SEATTLE - PLOTTING CHART



11.9. AFFORDABLE HOUSING INVENTORY – SELECTING HOUSING TYPE

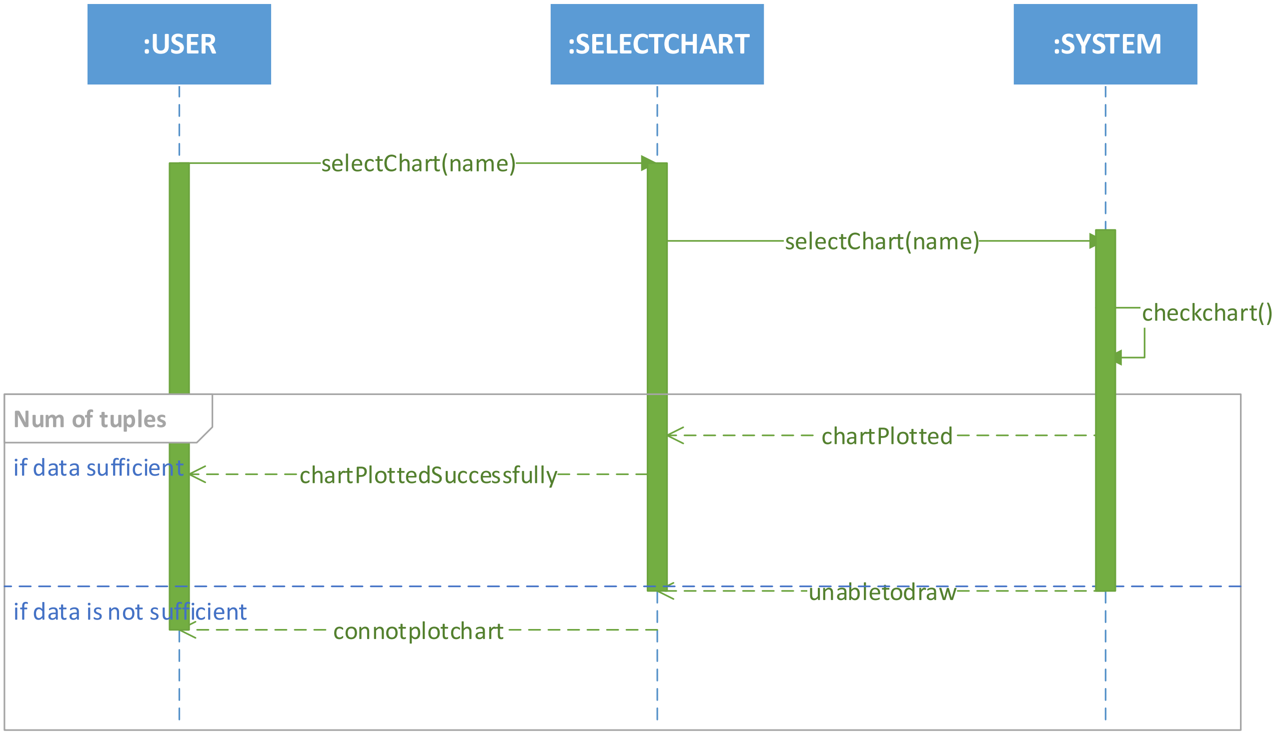
Assuming Main Page is already loaded

System is treated as a block



User can Filter the dataset as per his choice

11.10. AFFORDABLE HOUSING INVENTORY – PLOTTING GRAPH



12. DESIGN PATTERNS:

12.1 FACTORY METHOD DESIGN PATTERN: TO PLOT CHART

The Factory Method Design pattern defines an interface for creating and object, but lets the subclasses decide which class to instantiate. Factory Method lets a class defer instantiation to subclasses.It defines an interface for creating an object, but leaves the choice of its type to the subclasses, creation being deferred at run-time. Factory pattern is one of the most used design patterns in Java. This type of design pattern comes under creational pattern as this pattern provides one of the best ways to create an object.

In our scenario, On Plot Chart module, we have implemented factory pattern. we select from the multiple available chart options, as we call the createChart we pass the type of chart, and based on the type of chart, we instantiate the correct concrete class and assign it to the chart instance variable. Note that each chart will implement the Chart Interface. The concrete factories inherit the Factory method through the chart operation which is the base class for the chartConfig.

All the methods are provided by the createChart method which is super class acting as an interface for them. This is the only class referring to the concrete class Payment (abstract).

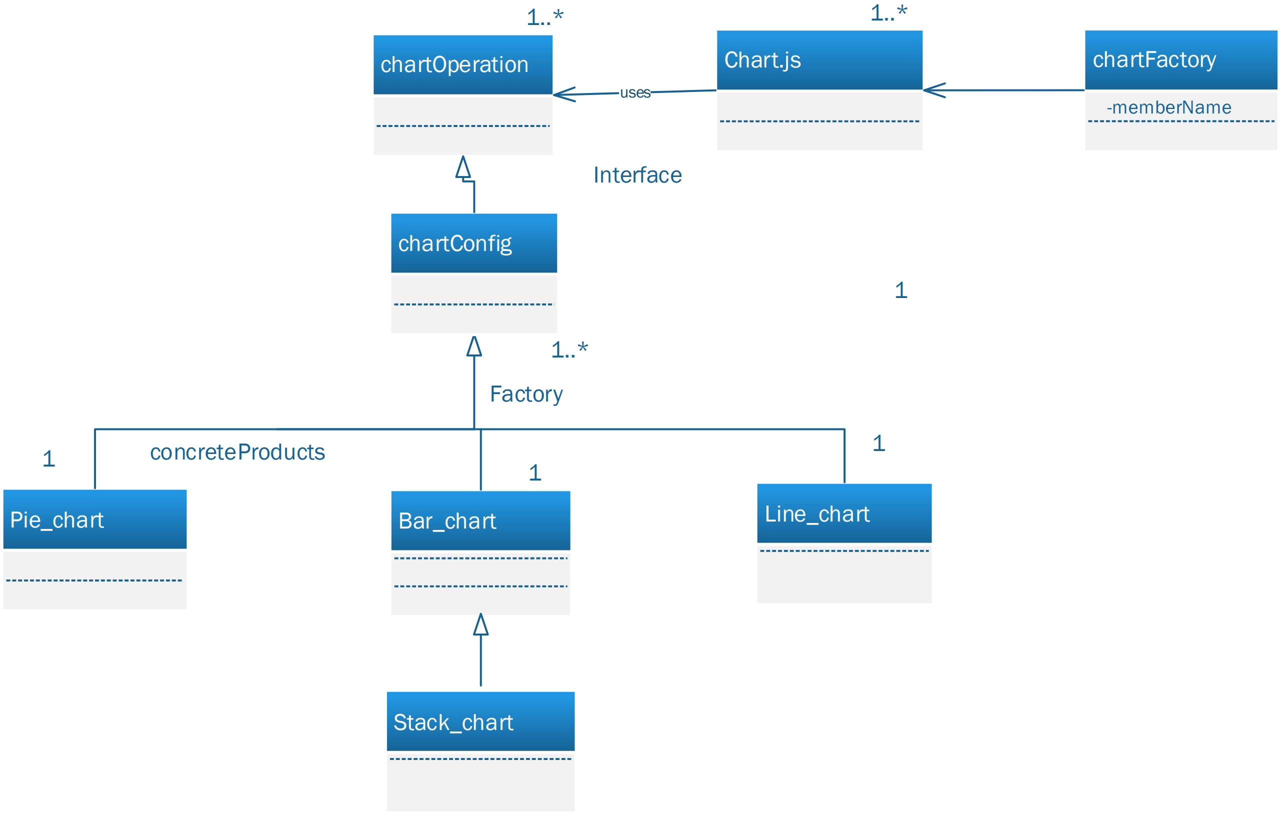


Fig: Factory Method Design Pattern

12.2 SINGLETON DESIGN PATTERN:DATASET

Singleton pattern is one of the simplest design patterns in Java. This type of design pattern comes under creational pattern as this pattern provides one of the best ways to create an object. This pattern involves a single class which is responsible to create an object while making sure that only single object gets created. This class provides a way to access its only object which can be accessed directly without need to instantiate the object of the class.

The Singleton Design Pattern ensures a class has only one instance, and provides a global point of access to it. This involves a single class which instantiates to create an object, making sure that only single object gets created. It provides a way to access its only object which can be accessed directly without need to instantiate the object of the class. This is useful when exactly one object is needed to coordinate actions across the system. The concept is sometimes generalized to systems that operate more efficiently when only one object exists, or that restrict the instantiation to a certain number of objects.

In our scenario, the ***Dataset*** class creates only one instance of a dataset for further operations such as data load, filter and chart plotting. This provides a global point of access to it. Using the variable Dataset, which is of the json objects, the entire system can know that whether the datasetObjects has been loaded to the dataframe to plot chart or not.

This class has made sure that the **s** method allows us to instantiate the DataSet class singlehandedly and to return an instance of it. Hence, the Singleton Design Pattern is used in this case.

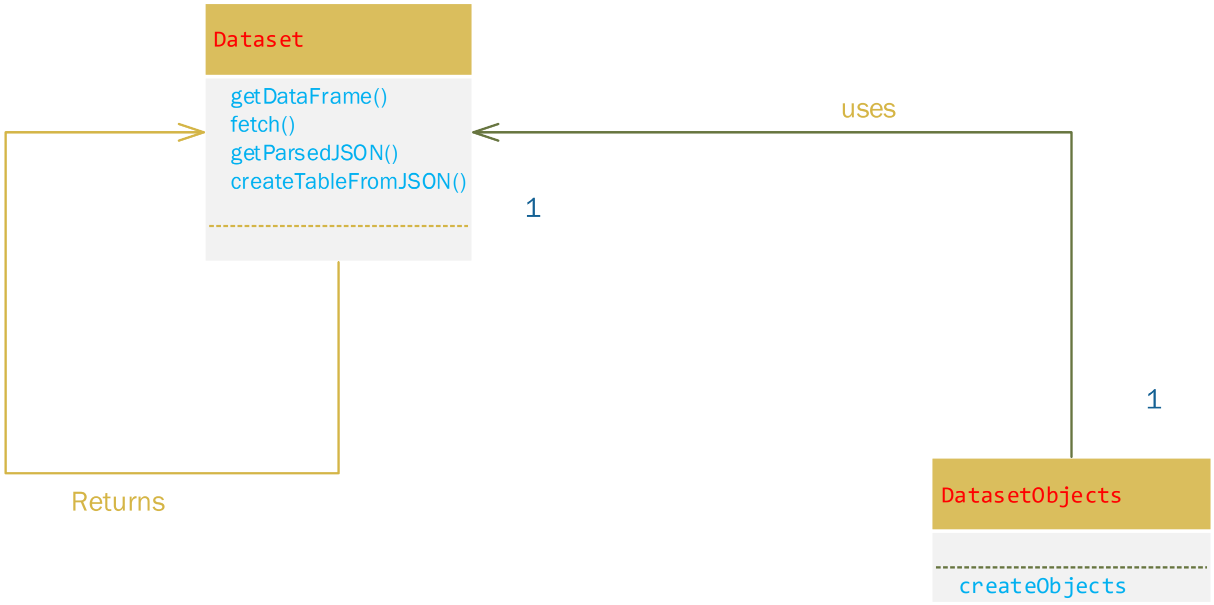


Fig: Singleton Design Pattern: DataSet

12.3 DECORATOR DESIGN PATTERN:

Decorator pattern allows a user to add new functionality to an existing object without altering its structure. This type of design pattern comes under structural pattern as this pattern acts as a wrapper to existing class.

This pattern creates a decorator class which wraps the original class and provides additional functionality keeping class methods signature intact.

This comes under structural pattern as it acts as a wrapper to existing class. Decorator Design pattern creates a decorator class which wraps the original class and provides additional functionality keeping class methods signature intact.

In our scenario, we have used chartDecorator class to add the background color to the charts, which won't alter the design of the chart but provides an additional functionality applyBackgroundcolor as a decorating feature.

The BarChartDecorator, PieChartDecorator, LineChartDecorator etc are the decorator subclass which implements the ChartDecorator class to provide the additional background color for the charts.

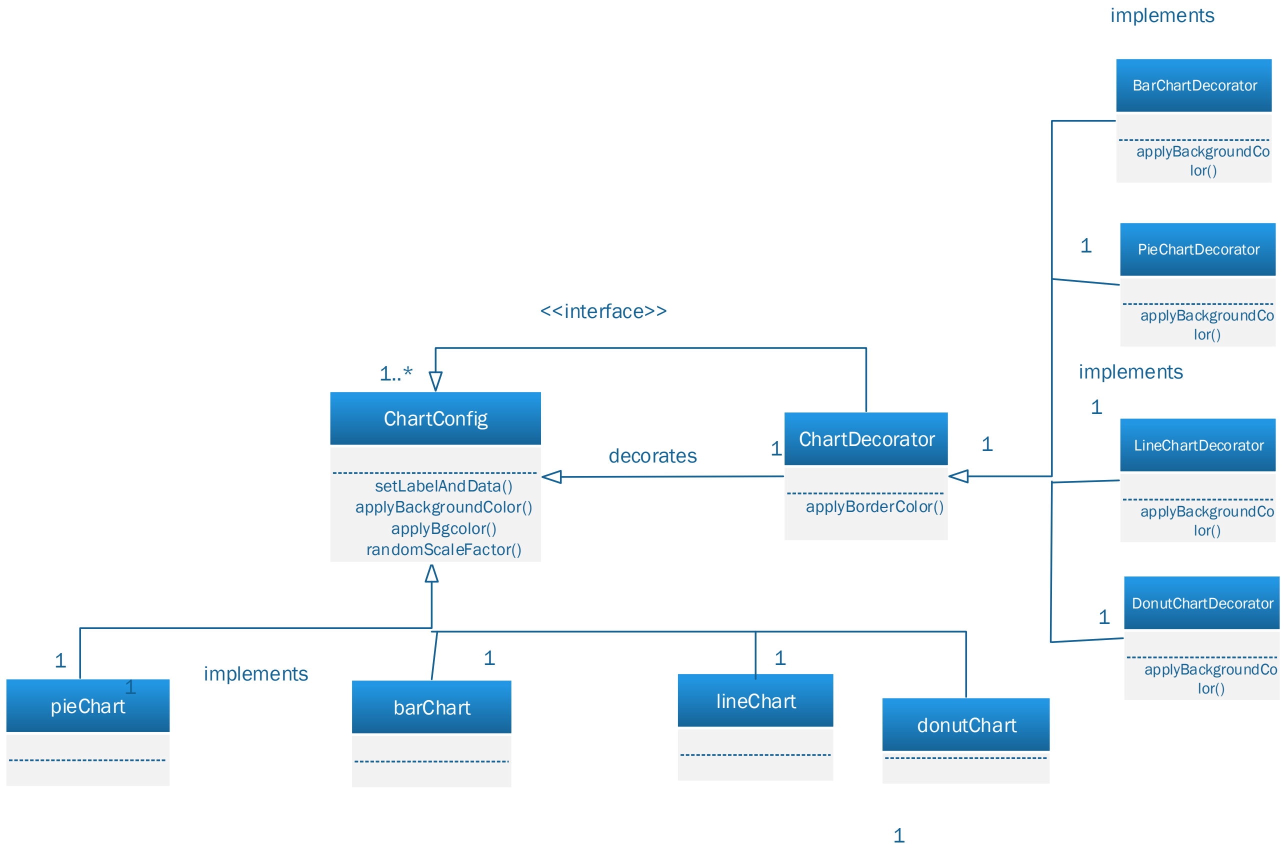


Fig: Decorator Design Pattern: ColorConfiguration

12.4 MVC DESIGN PATTERN:

MVC Pattern stands for Model-View-Controller Pattern. This pattern is used to separate application's concerns.

- **Model** - Model represents an object carrying data. It can also have logic to update controller if its data changes.
- **View** - View represents the visualization of the data that model contains.
- **Controller** - Controller acts on both model and view. It controls the data flow into model object and updates the view whenever data changes. It keeps view and model separate.

It divides a given application into three interconnected parts to separate internal representations of information from the ways that information is presented to and accepted from the user. The files via which this pattern has been implemented is listed below:

- HousingMain.js-View represents the visualization of the data that model contains.
- ModelUtility.js: Model represents an object. It can also have logic to update controller if its data changes.
- ControllerUtility.js: Controller acts on both model and view. It controls the data flow into model object and updates the view whenever data changes. It keeps view and model separate.

13. CONCLUSION:

The dashboard is developed and implemented using the design patterns with the help of ECMA6, DataFrame.js and Chart.js libraries which provides an interactive way of plotting chart. It is an efficient way of utilizing Java Design Patterns and OOPS concepts.

In future we can use 3D charts as an further enhancements.

***