DEMOGRAPHICS

Team 15

Bidari, Vishwanath (A20352720) (vbidari@hawk.iit.edu)

Kolanupaka,Saisruthi (A20388341) (skolanupaka@hawk.iit.edu)

MadbalGowda,Aishwarya (A20391651) (amadbalgowda@hawk.iit.edu)

Nanavati, Vinita (A20379862) (vnanavati@hawk.iit.edu)

Vinay, Pavithra (A20369869) (pvinay@hawk.iit.edu)

CSP 586-Software Modeling and Development with UML

1. **Overview**:

Demographics is study of populations, with reference to size and density, mortality, age distribution, and other vital statistics and the integration of all these with social and economic conditions.

Goal: To implement an object oriented charting Dashboard for Demographic data sets. This helps user in visualization of data in the form of charts (Pie chart, Bar chart, Doughnut chart, Stack chart, Line chart). User can view the census related data of different cities in the form of these charts.

Object oriented Dashboard is modelled and designed using UML and implemented using ES6 Javascript along with chart.js charting library and as for a DataFrame object dataframe-js library is used for filtering/slicing/dicing the datasets.

2. **Requirements/Features List :**

1. Select a dataset from available datasets

   Datasets:

   - Houston Census Demographics
   - New York Census Demographics
   - Washington Census Demographics
   - Vegas Census Demographics
   - Baton Rouge Census Demographics

2. Read data from CSV File
3. Load data into data frame object using Dataframe.js
4. Display loaded dataset in tabular format
5. Select data from dataset

   Data selection can be done dynamically based on the selected dataset

6. Filter data from selected columns of dataset for charting

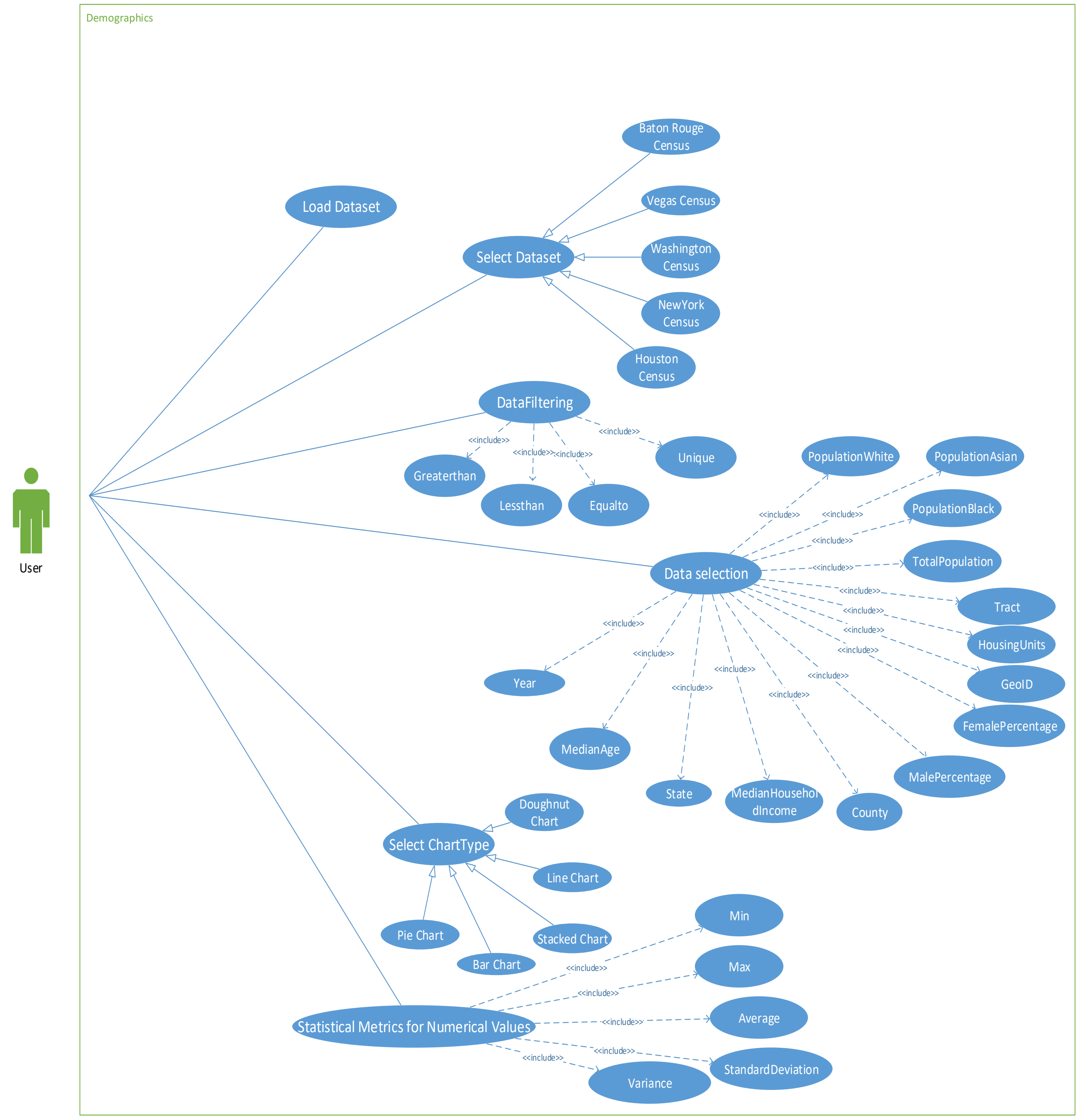
Data filtering is done based on filtering criteria “greater than”, ”less than”, “equal to”, “unique row values” of different columns selected using selection.

7. Select chart type such as Line, Bar, Pie, Stack, Doughnut
8. Create complex analytical queries to plot the chart
9. Plot data as chart using Chart.js

3. Use Cases

- User selects Houston Census dataset
- User selects New York Census dataset
- User selects Washington Census dataset
- User selects Vegas Census dataset
- User selects Baton Rouge Census dataset
- User loads the dataset
- User selects the required columns based on the dynamically displayed data table
- User selects “ greater than” as filtering criteria
- User selects “less than” as filtering criteria
- User selects “equal to” as filtering criteria
- User selects “unique row values” as filtering criteria
- User performs query to get “Min” statistical values for numerical data
- User performs query to get “Max” statistical values for numerical data
- User performs query to get “Average” statistical values for numerical data
- User creates complex analytical queries based on the column names and values in dataset
- User selects chart type
- User plots the chart

3.1 Use Case Diagram:



3.2 Use Cases Fully dressed:

i) Load Dataset

Conditions	**Documentation**
Use case Name	Load Dataset
Scope	Demographics
Level	User Goal Level
Primary Actor	User
Stakeholders and Interests	User wants to understand graphical representation of datasets
Preconditions	• User must have access to all datasets • DataSet should be in correct format (CSV)
Success guarantee (Post Condition)	Data is displayed in tabular format
Main Success Scenario	• Select the DataSet from the list of available dataset namely Houston Census ,Vegas Census ,New York Census, Baton Rouge Census, Washington Census in CSV format. • User requests for census of one city from five different cities in USA. The data is loaded and displayed on window in tabular format
Extensions (or Alternative Flows):	none
Special Requirements	System should be able to verify DataSet File Format.
Technology and Data Variation List	Chart.js, dataframe.js, javascript, html, Firefox.
Frequency of Occurrence	High as we need to load dataset every time we need to access a different dataset
Open Issues	none

ii) Data Selection

Conditions	Documentation
Use case Name	Data Selection
Scope	Demographics
Level	User Goal Level
Primary Actor	User
Stakeholders and Interests	User wants to understand graphical representation of datasets and view column specific data.
Preconditions	• Data should be loaded • Data should be displayed I tabular format. • Checkboxes with column names to make selection should be available dynamically based on the selected dataset
Success guarantee (Post Condition)	Selected columns should be displayed in tabular format
Main Success Scenario	• Select the DataSet from the list of available dataset namely Houston Census ,Vegas Census ,New York Census, Baton Rouge Census, Washington Census in CSV format • User requests for census of one city from five different cities in USA.The data is loaded and displayed on window as a table • User has an option to select the data i.e any required columns based on dynamically displayed tabular data such as tract, total population, black population, white population, Asian population, other population etc… • Only Selected columns should be displayed in the existing data table
Extensions (or Alternative Flows):	none
Special Requirements	User should be able to select all available columns dynamically based on loaded dataset
Technology and Data Variation List	Chart.js, dataframe.js, javascript, html, Firefox.
Frequency of Occurrence	High
Open Issues	none

iii) Data filtering for unique row values

Conditions	**Documentation**
Use case Name	Data filtering for unique row values
Scope	Demographics
Level	User Goal Level
Primary Actor	User
Stakeholders and Interests	User wants to understand graphical representation of datasets and filter data based on distinct row values.
Preconditions	• Data should be displayed in tabular format. • Data selection is completed. • Checkbox with filtering options to select unique row values should be available.
Success guarantee (Post Condition)	Filtered data should be displayed in tabular format with unique row values
Main Success Scenario	• Select the DataSet from the list of available dataset namely Houston Census ,Vegas Census ,New York Census, Baton Rouge Census, Washington Census in CSV format • User requests for census of one city from five different cities in USA.The data is loaded and displayed on window in tabular format. • User has an option to select the data i.e any required columns based on dynamically displayed tabular data such as tract,total population,black population,white population,Asian population,other population etc… • Only Selected columns should be displayed in the existing table. • User should be able to select unique row values • Filtered data should be displayed in tabular format
Extensions (or Alternative Flows):	none
Special Requirements	User should be able to filter available rows dynamically based on loaded and selected dataset.
Technology and Data Variation List	Chart.js, dataframe.js, javascript, html, Firefox
Frequency of Occurrence	High
Open Issues	none

iv) Data filtering with “greater than” as filter criteria

Conditions	**Documentation**
Use case Name	Data filtering for Greater than values
Scope	Demographics
Level	User Goal Level
Primary Actor	User
Stakeholders and Interests	User wants to understand graphical representation of datasets and filter data based on greater than values for numerical data
Preconditions	• Data should be displayed in tabular format. • Data selection is completed. • Checkbox with filtering options to select unique row values should be available • Filters for unique row values should be selected
Success guarantee (Post Condition)	Filtered data should be displayed in tabular format
Main Success Scenario	• Select the DataSet from the list of available dataset namely Houston Census ,Vegas Census ,New York Census, Baton Rouge Census, Washington Census in CSV format • User requests for census of one city from five different cities in USA.The data is loaded and displayed on window in tabular format. • User has an option to select the data i.e any required columns based on dynamically displayed tabular data such as tract, total population, black population, white population, Asian population, other population etc… • Only Selected columns should be displayed in the existing table. • User should be able to select unique row values • User should have options to select greater than • User should be able to select values for greater than from the displayed drop down • Filtered data should be displayed in tabular format

Extensions (or Alternative Flows):	• Select the DataSet from the list of available dataset namely Houston Census ,Vegas Census ,New York Census, Baton Rouge Census, Washington Census in CSV format • User requests for census of one city from five different cities in USA.The data is loaded and displayed on window in tabular format. • User has an option to select the data i.e any required columns based on dynamically displayed tabular data such as tract,total population,black population,white population,Asian population,other population etc… • Only Selected columns should be displayed in the existing table. • User should be able to select unique row values • User should have options to select "less than" or "equal to" • User should be able to select values for "less than" or "equal to" from the displayed drop down • Filtered data should be displayed in tabular format
Special Requirements	User should be able to filter available rows dynamically based on loaded and selected dataset.
Technology and Data Variation List	Chart.js, dataframe.js, javascript, html, Firefox
Frequency of Occurrence	Moderate
Open Issues	none

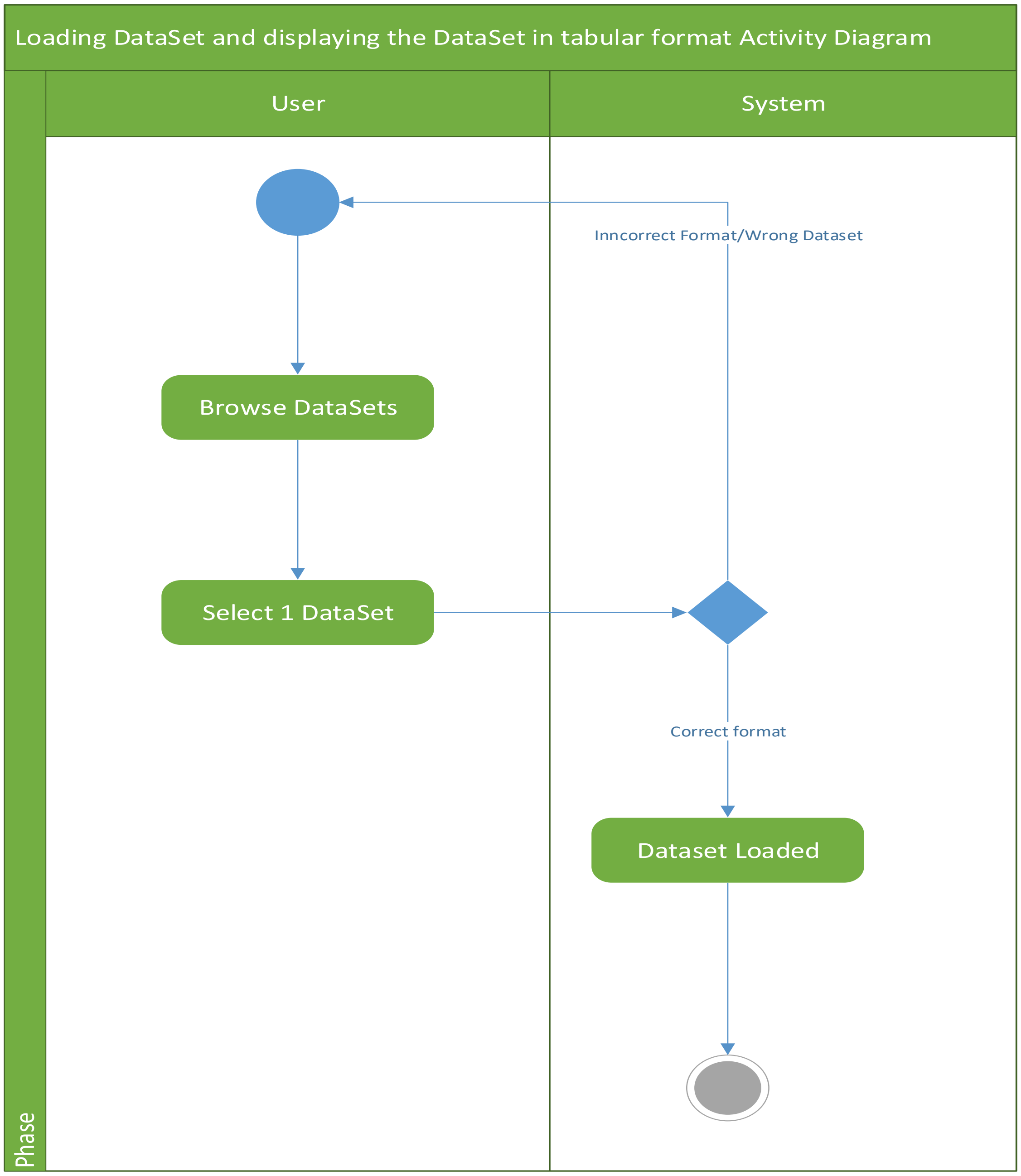
v) Select Chart type(s)

Conditions	Documentation
Use case Name	Select chart type
Scope	Demographics
Level	User Goal Level
Primary Actor	User
Stakeholders and Interests	User wants to understand graphical representation of datasets
Preconditions	• Data should be displayed in tabular format • Data selection (selection of required columns) should be done • Data filtering according to required values should be done • Checkboxes with different chart types should be loaded dynamically
Success guarantee (Post Condition)	• Selected chart type should be displayed (pie, bar, stacked, line, Doughnut)
Main Success Scenario	• Select the DataSet from the list of available dataset namely Houston Census ,Vegas Census ,New York Census, Baton Rouge Census, Washington Census in CSV format • User requests for census of one city from five different cities in USA.The data is loaded and displayed on window in tabular format • User has an option to select the data i.e any required columns based on dynamically displayed tabular data such as tract,total population,black population,white population,Asian population,other population etc… • Only Selected columns should be displayed in the existing table • User should be able to select required filters • Chart types are displayed dynamically after filtering • User selects the desired chart • User must be able to select x-axis, y-axis,z-axis(optional) based on selected columns and filter

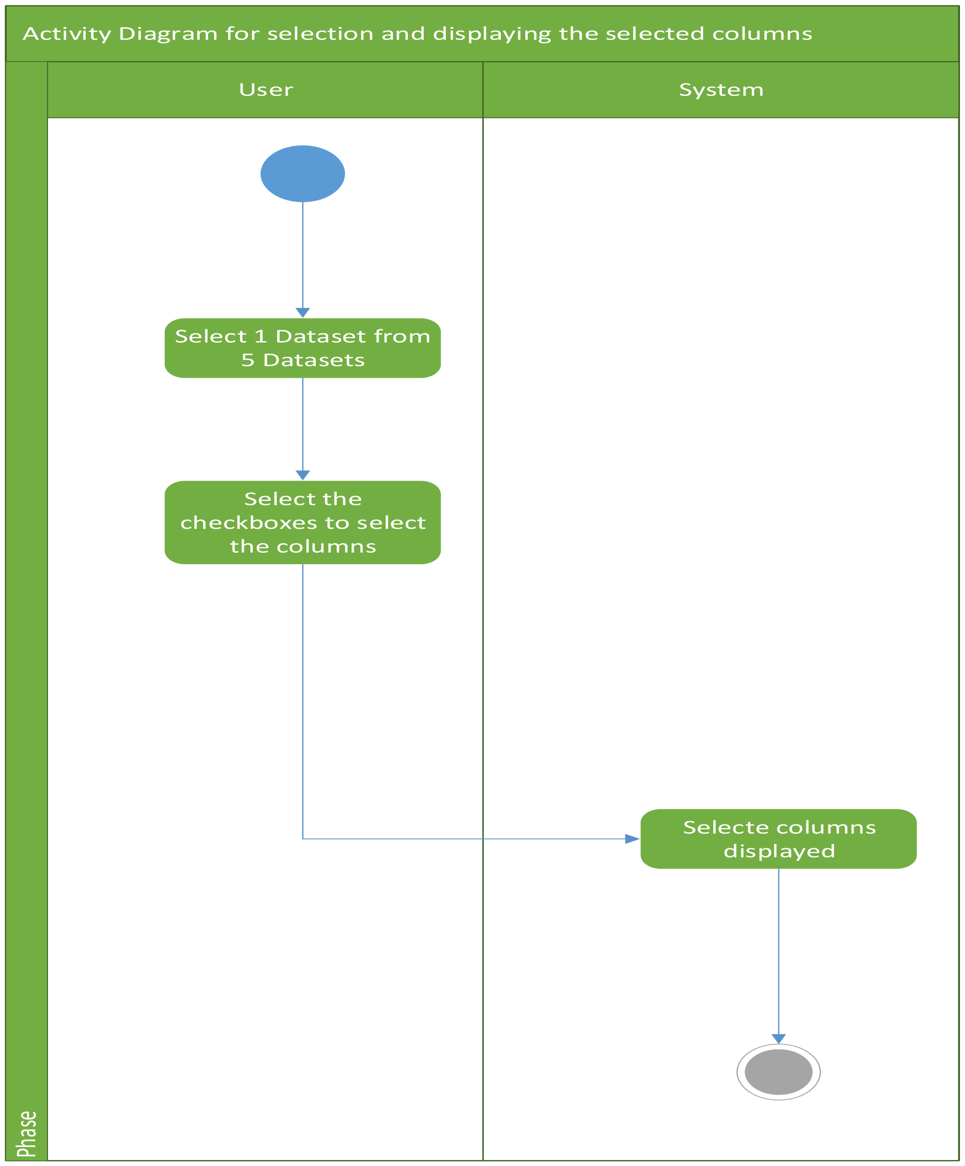
	values • User selects DrawCharts • Display the selected chart type
Extensions (or Alternative Flows):	none
Special Requirements	User should be able select the desired chart
Technology and Data Variation List	Chart.js, dataframe.js, javascript, html, Firefox
Frequency of Occurrence	Moderate
Open Issues	none

4. Activity Diagrams:

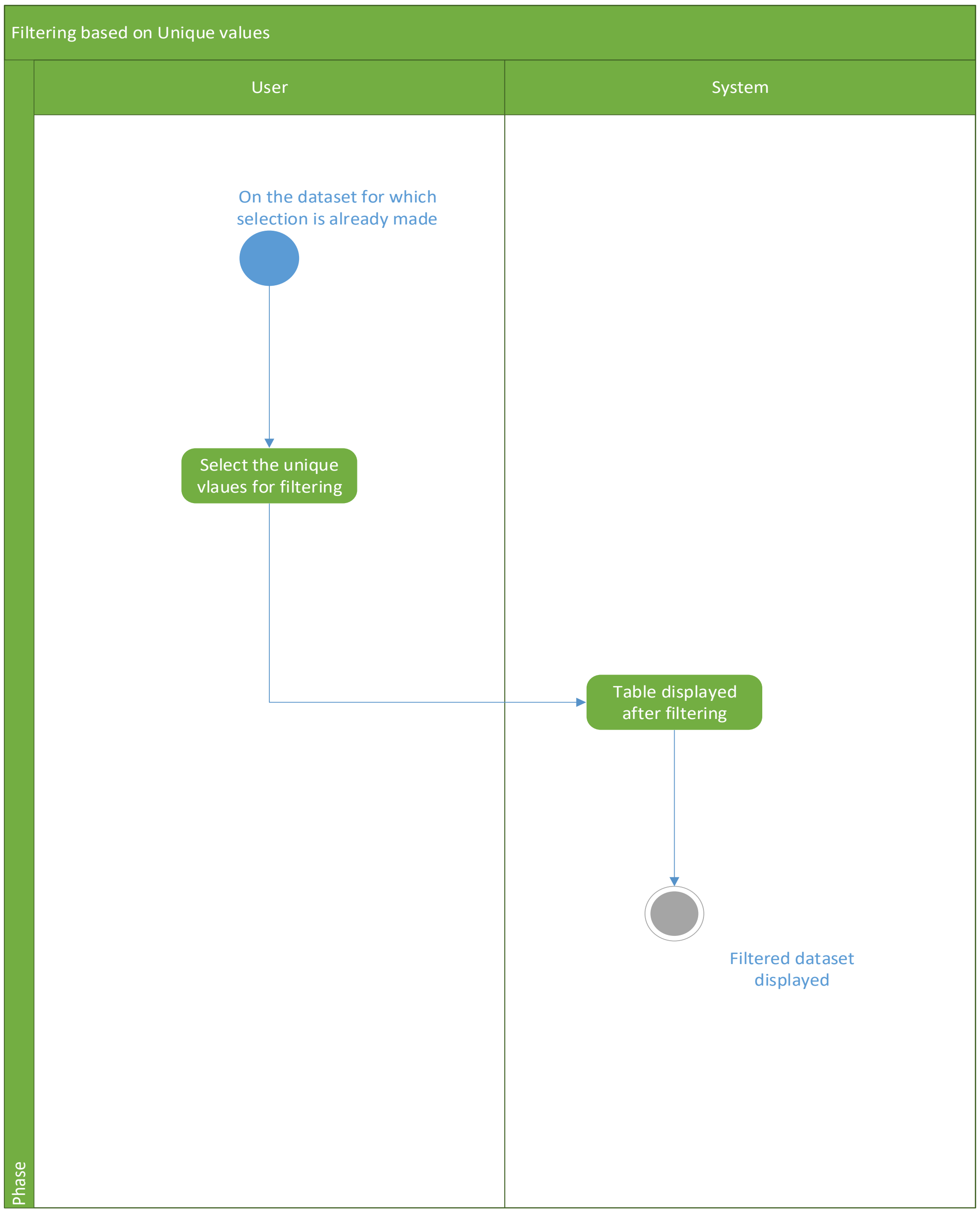
i) The user browses the datasets, selects the datasets and if the format for dataset selected is correct then the dataset will be displayed in tabular format.


Loading DataSet and displaying the DataSet in tabular format Activity Diagram
User
System
Phase
Browse DataSets
Select 1 DataSet
Inncorrect Format/Wrong Dataset
Correct format
Dataset Loaded

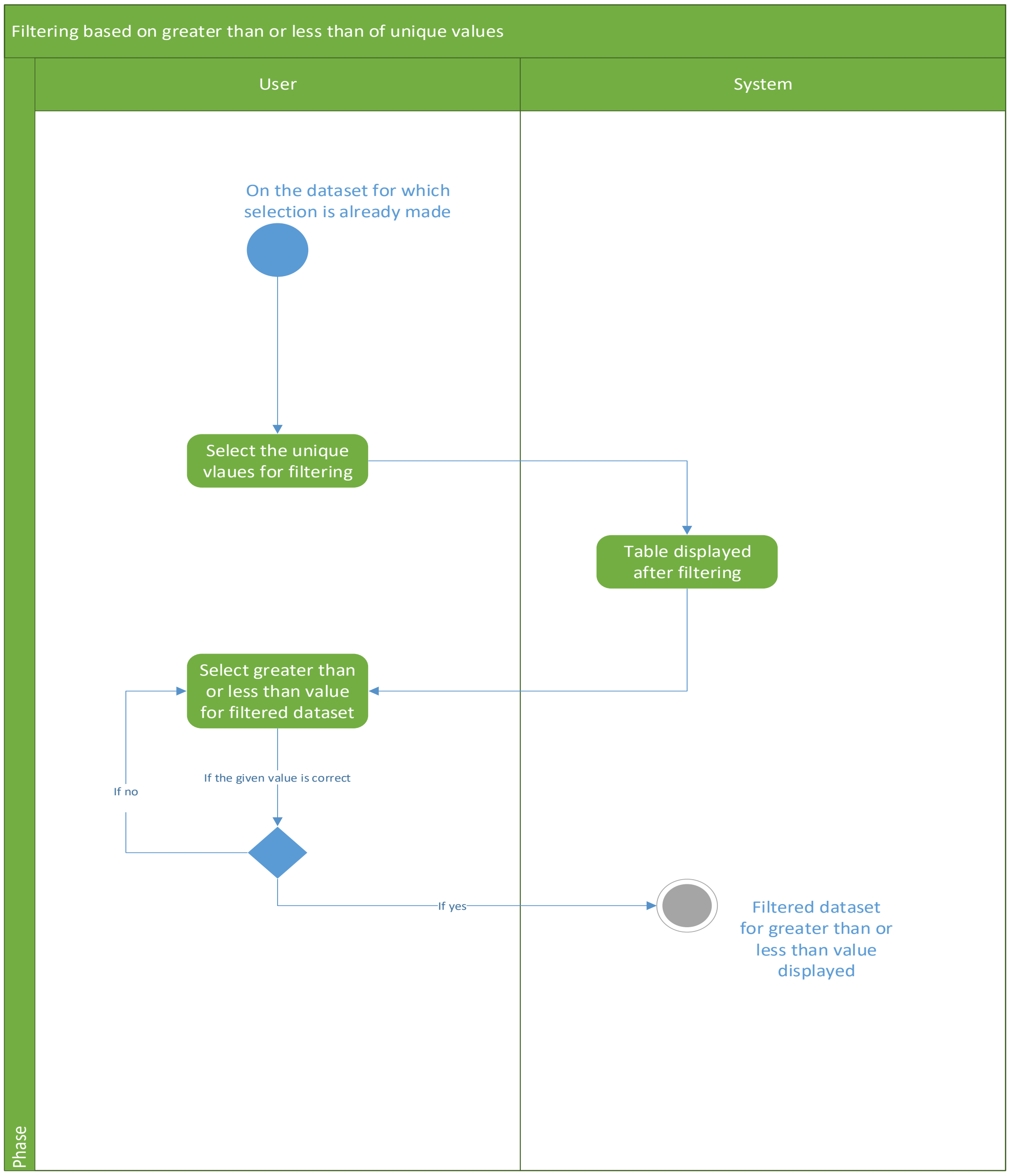
ii) Activity Diagram for selection of columns after the dataset is displayed


Activity Diagram for selection and displaying the selected columns
User
System
Phase
Select 1 Dataset from 5 Datasets
Select the checkboxes to select the columns
Selecte columns displayed

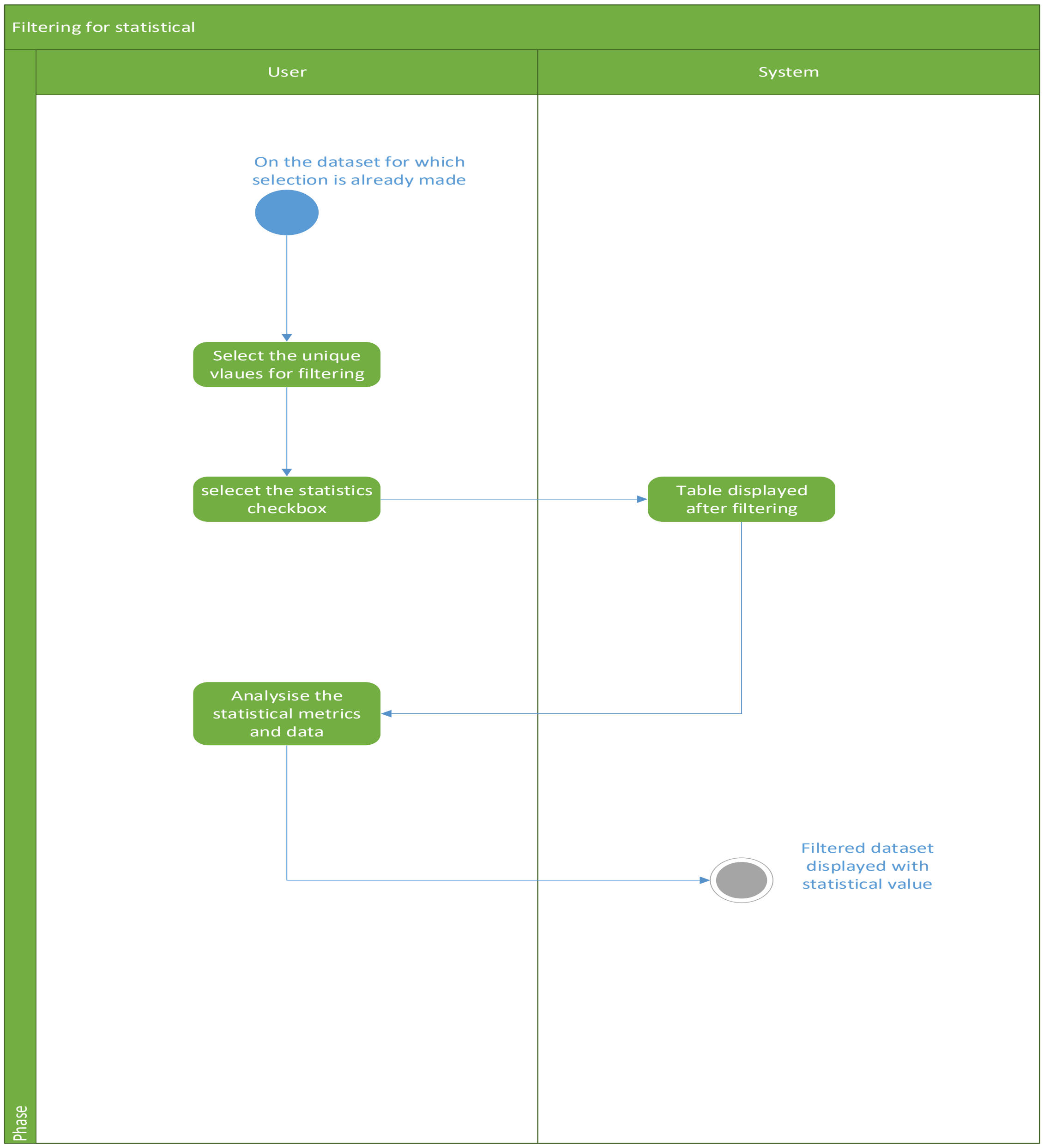
iii) Activity diagram for filtering based on unique entities


Filtering based on Unique values
User
System
On the dataset for which selection is already made
Select the unique vlaues for filtering
Table displayed after filtering
Filtered dataset displayed
Phase

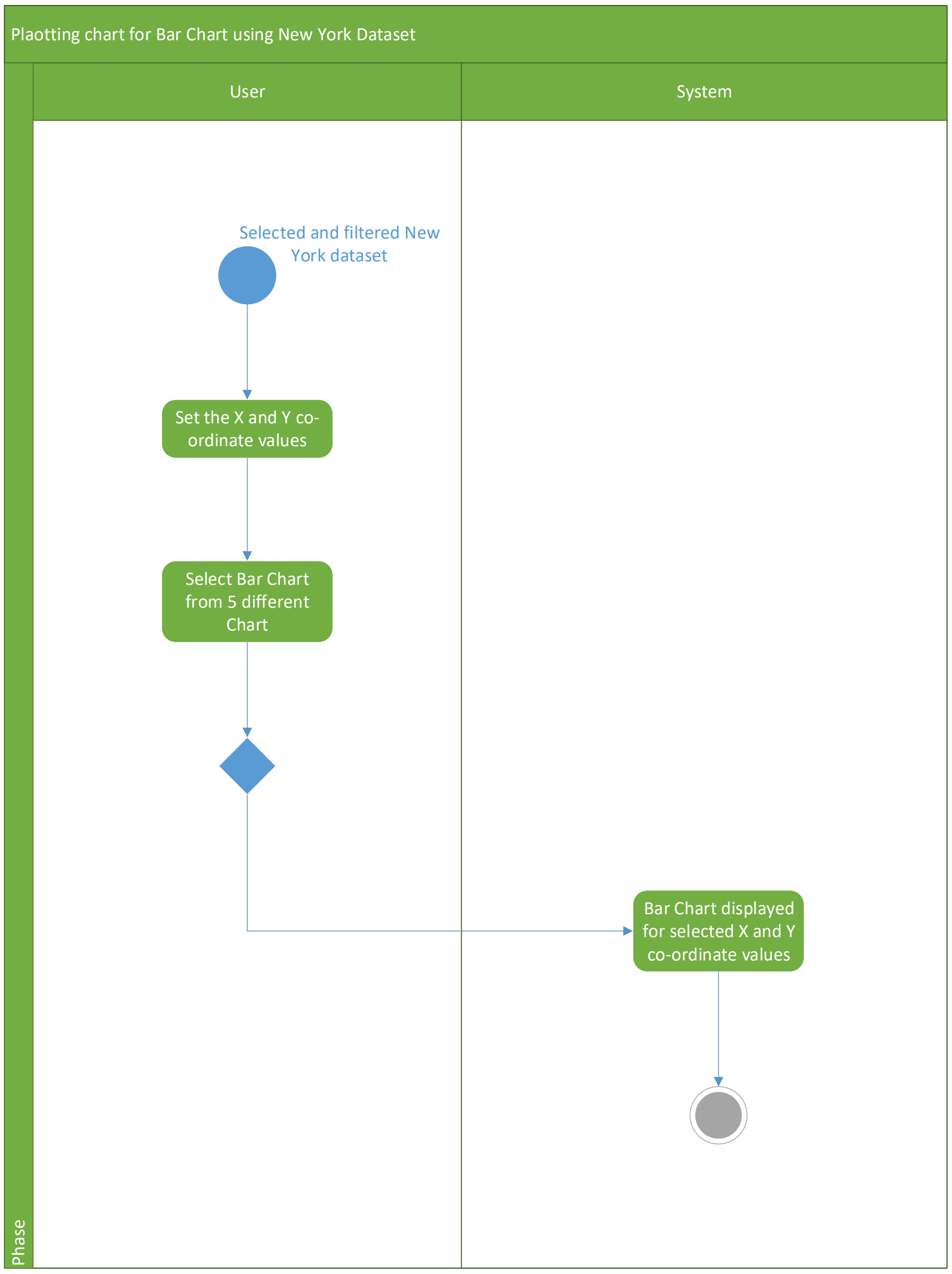
iv) Activity Diagram for filtering based on greater than or less than value of filtered dataset


Filtering based on greater than or less than of unique values
User
System
Phase
On the dataset for which selection is already made
Select the unique vlaues for filtering
Table displayed after filtering
Select greater than or less than value for filtered dataset
If the given value is correct
If no
If yes
Filtered dataset for greater than or less than value displayed

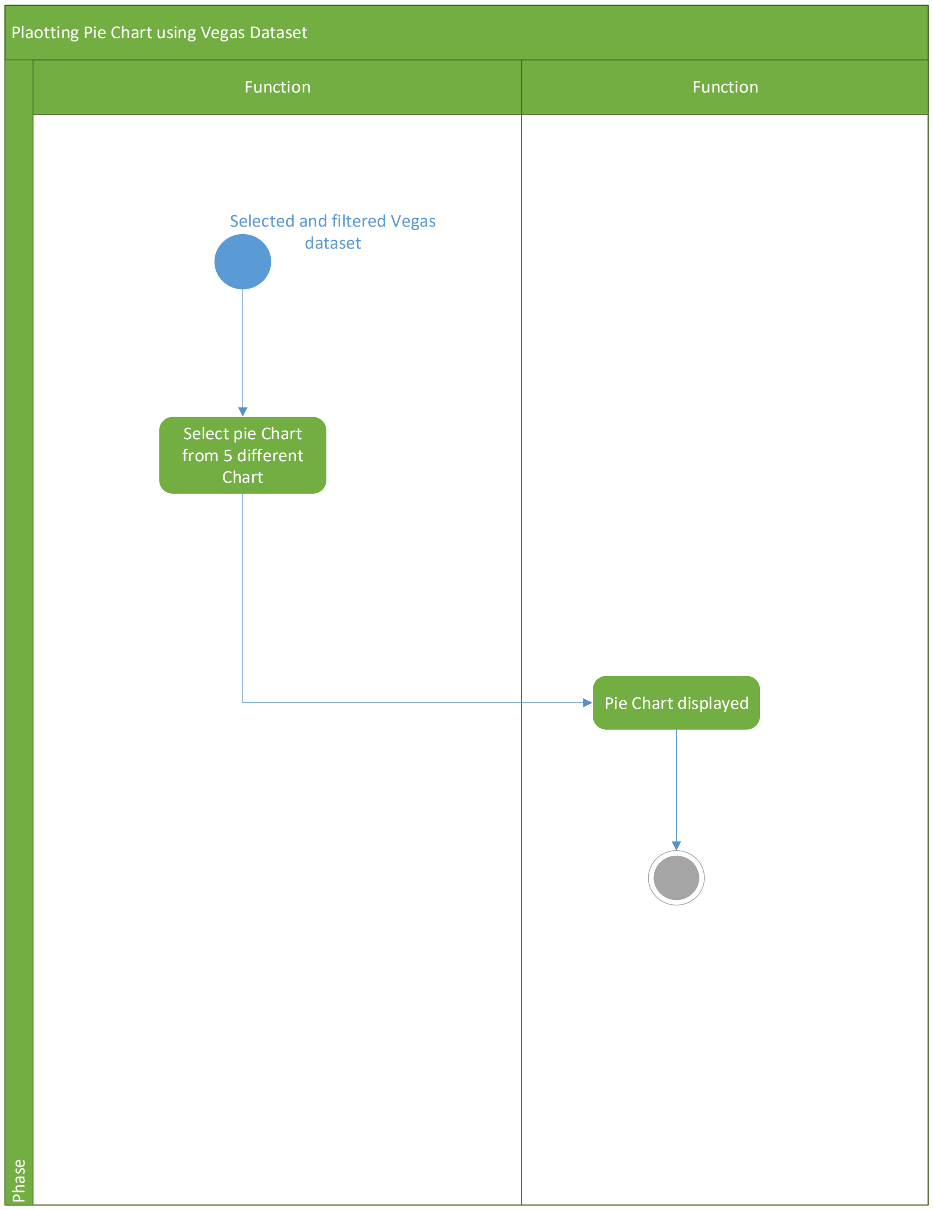
v) Activity Diagram for filtering based on Statistical Metrics of filtered dataset


Filtering for statistical
User
System
Phase
On the dataset for which selection is already made
Select the unique vlaues for filtering
selecet the statistics checkbox
Table displayed after filtering
Analysise the statistical metrics and data
Filtered dataset displayed with statistical value

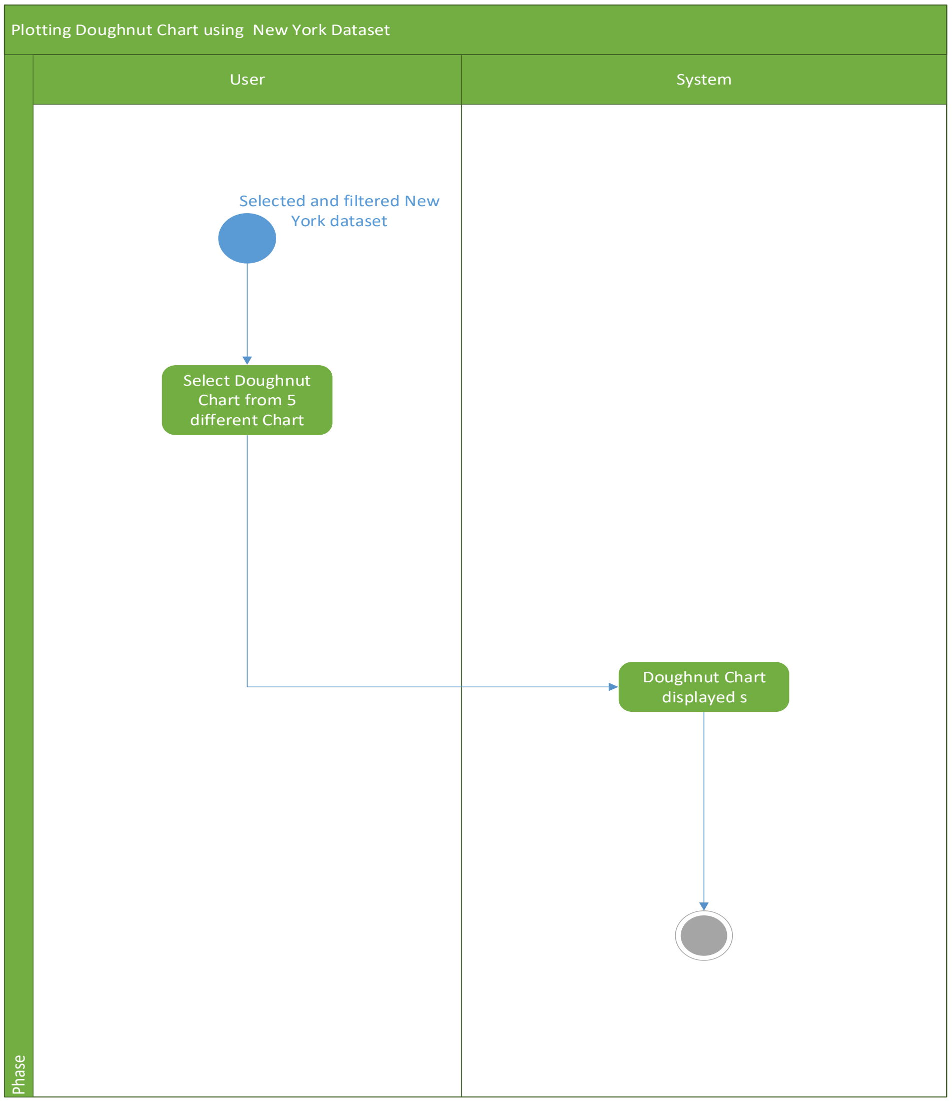
vi) Activity Diagram for bar chart using New York Dataset


Plaotting chart for Bar Chart using New York Dataset
User
System
Phase
Selected and filtered New York dataset
Set the X and Y co-ordinate values
Select Bar Chart from 5 different Chart
Bar Chart displayed for selected X and Y co-ordinate values

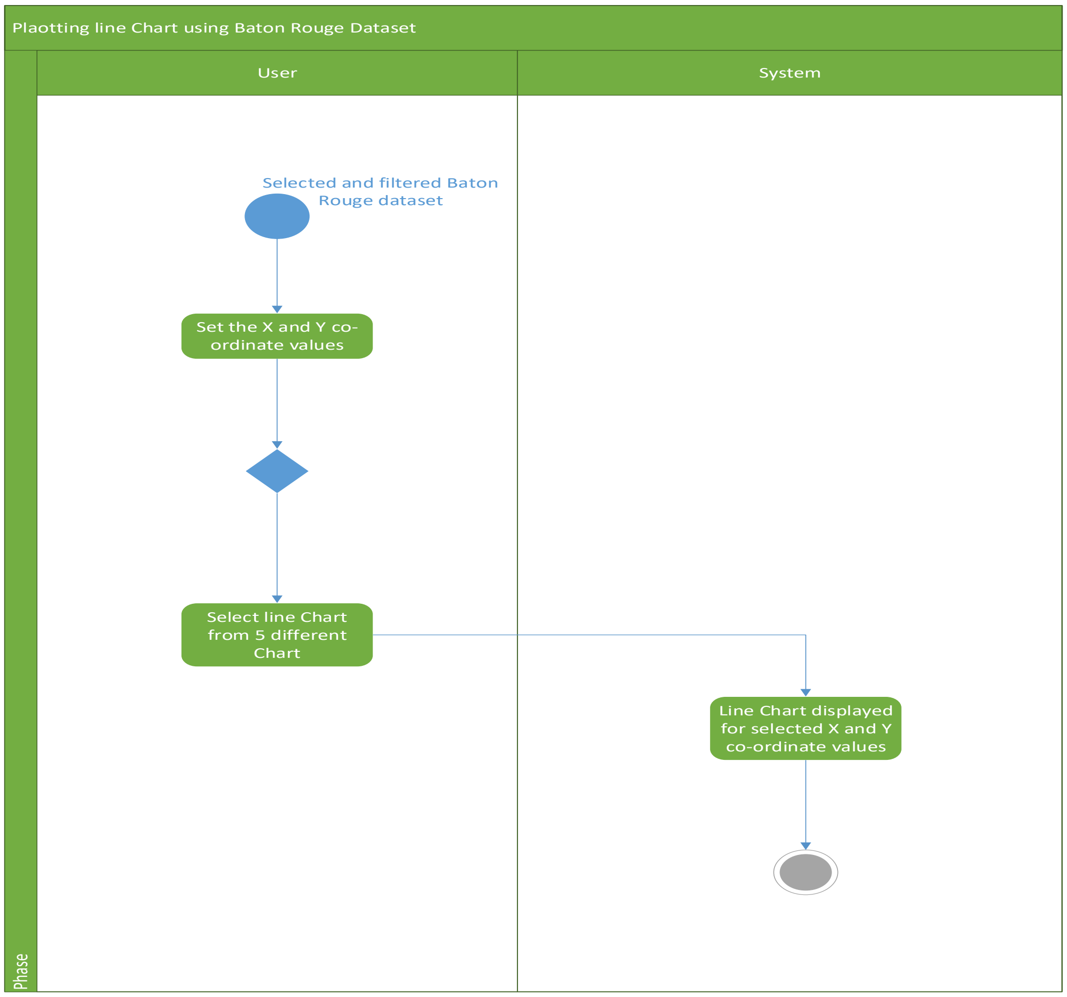
vii) Activity Diagram for pie chart using Vegas Dataset


Plaotting Pie Chart using Vegas Dataset
Function
Function
Phase
Selected and filtered Vegas dataset
Select pie Chart from 5 different Chart
Pie Chart displayed

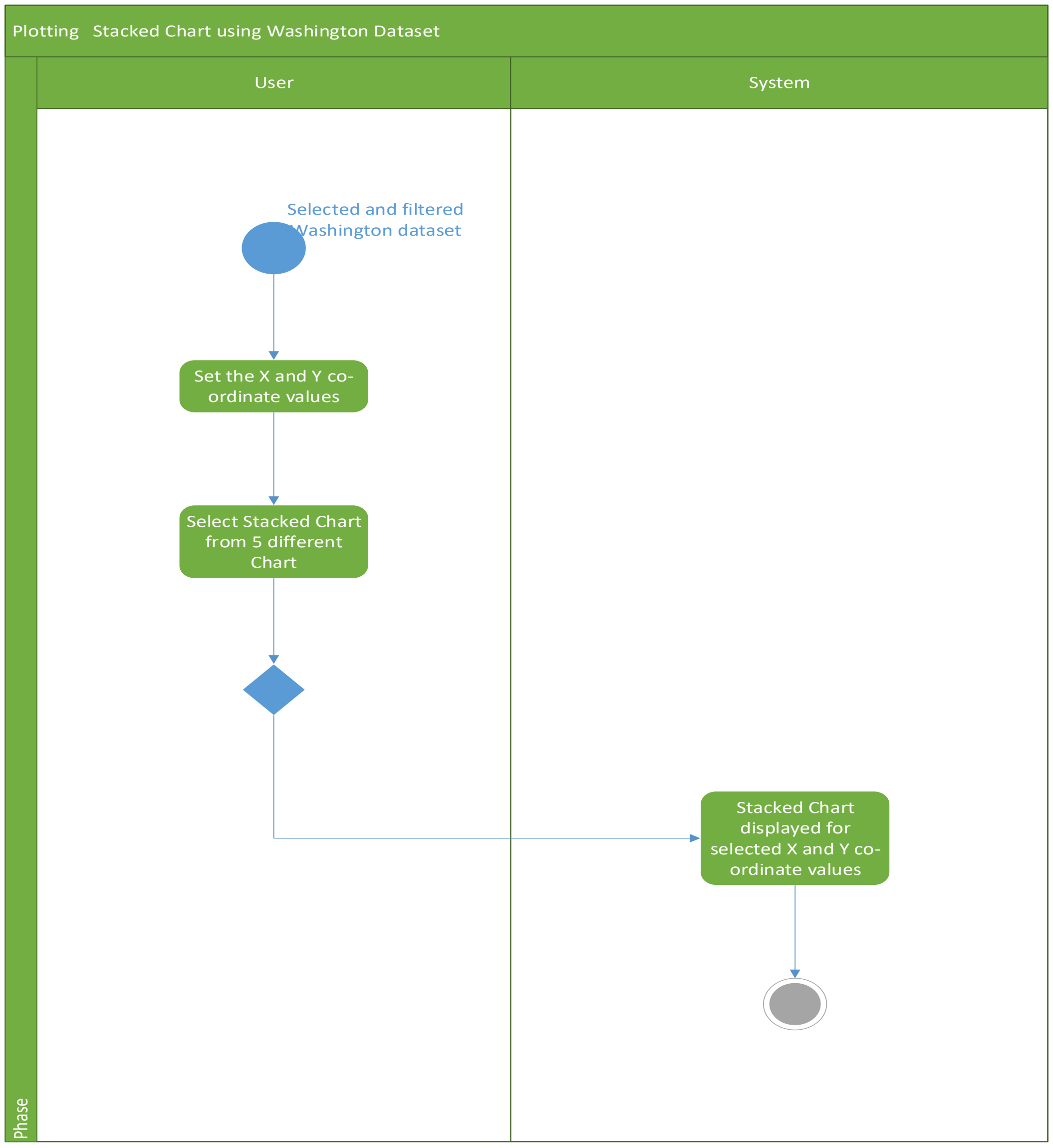
viii) Activity Diagram for Doughnut Chart using New York dataset


Plotting Doughnut Chart using New York Dataset
User
System
Phase
Selected and filtered New York dataset
Select Doughnut Chart from 5 different Chart
Doughnut Chart displayed s

ix) Activity Diagram for line chart using Baton Rouge Dataset


Plaotting line Chart using Baton Rouge Dataset
User
System
Phase
Selected and filtered Baton Rouge dataset
Set the X and Y co-ordinate values
Select line Chart from 5 different Chart
Line Chart displayed for selected X and Y co-ordinate values

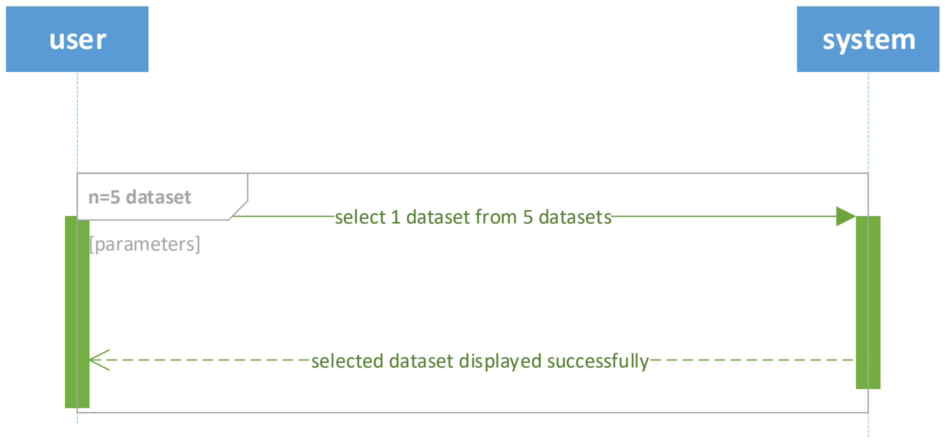
x) Activity Diagram Stacked chart using Washington Dataset


Plotting Stacked Chart using Washington Dataset
User
System
Phase
Selected and filtered Washington dataset
Set the X and Y co-ordinate values
Select Stacked Chart from 5 different Chart
Stacked Chart displayed for selected X and Y co-ordinate values

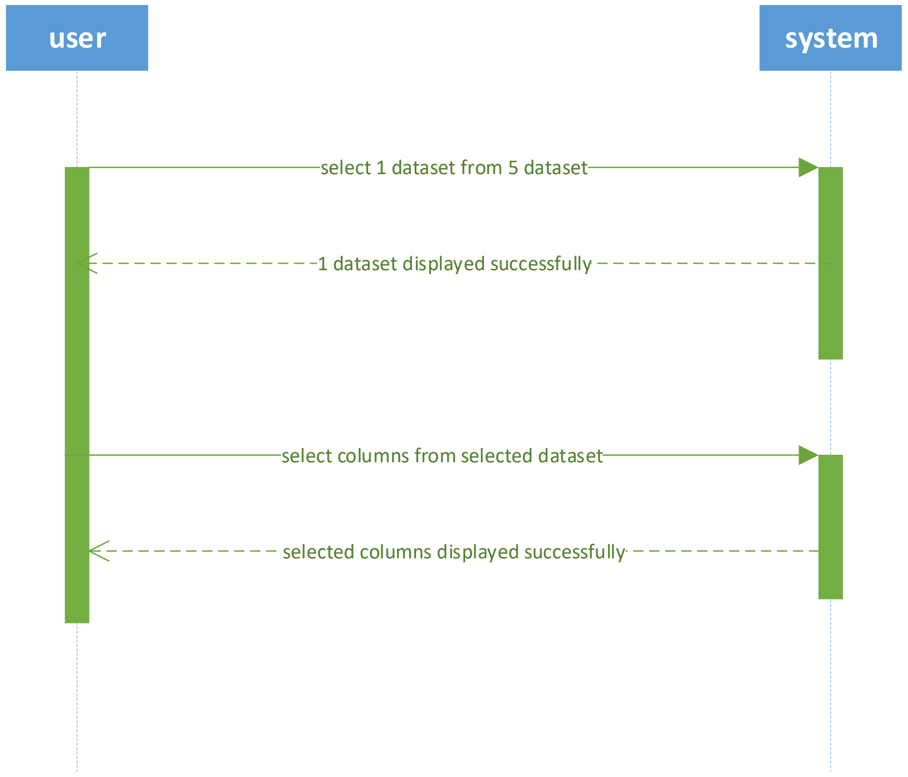
5. Sequence Diagrams:

5.1 System Sequence Diagrams

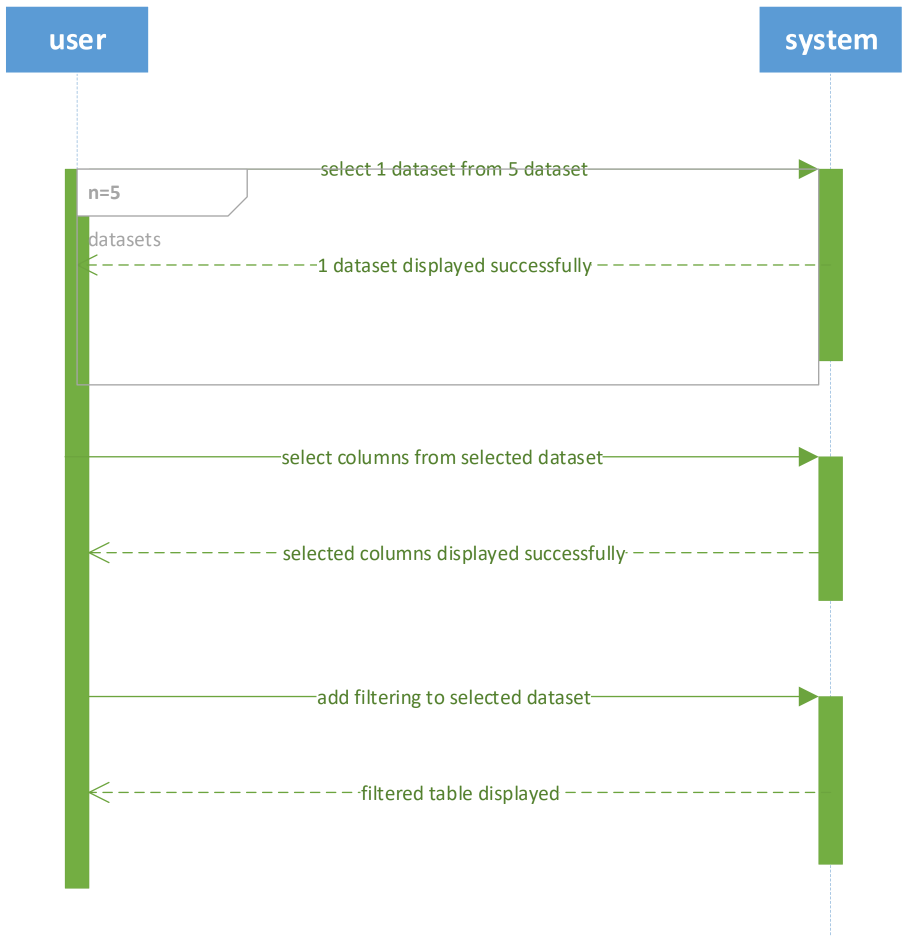
i) Sequence Diagram for Loading Dataset


user
system
n=5 dataset
[parameters]
select 1 dataset from 5 datasets
selected dataset displayed successfully

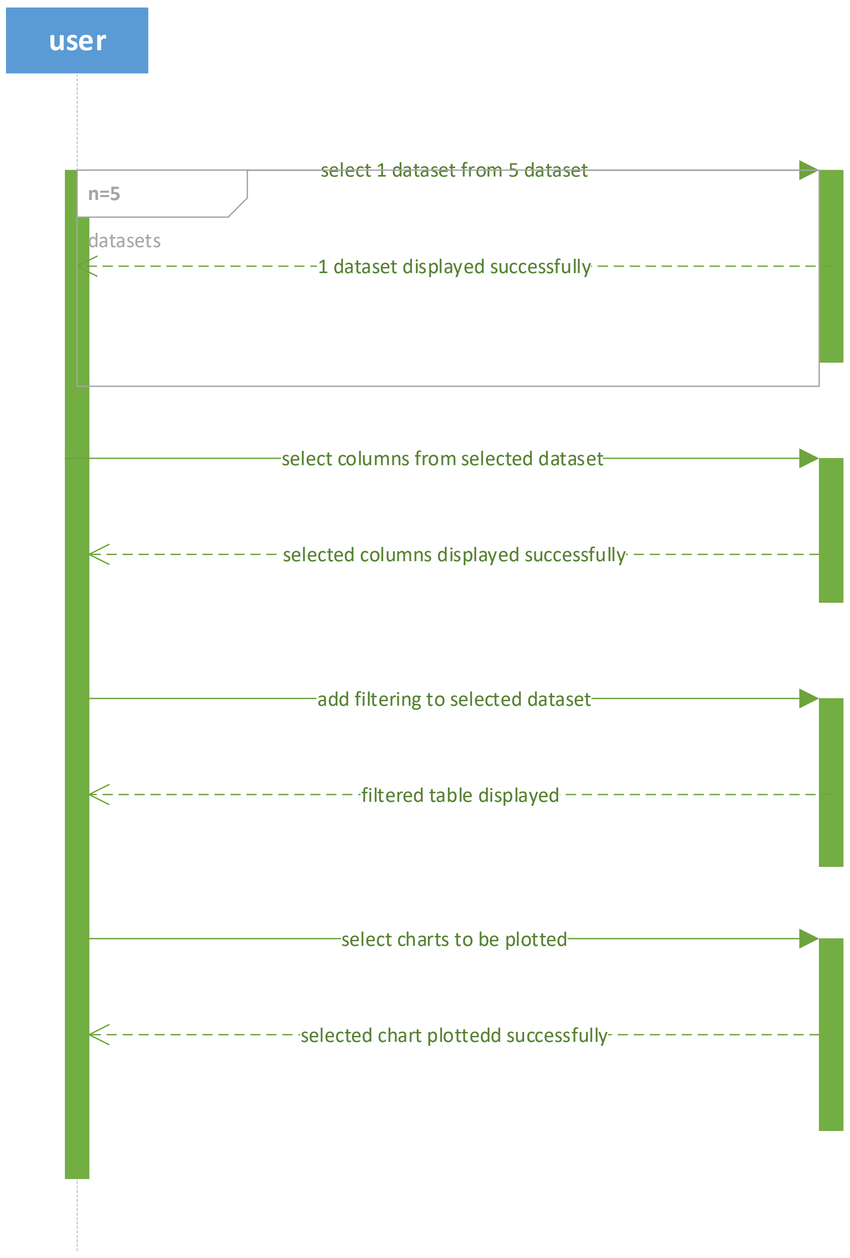
ii) Sequence Diagram for selecting column of the loaded dataset


user
system
select 1 dataset from 5 dataset
1 dataset displayed successfully
select columns from selected dataset
selected columns displayed successfully

iii) Sequence diagram for Filtering of selected Dataset

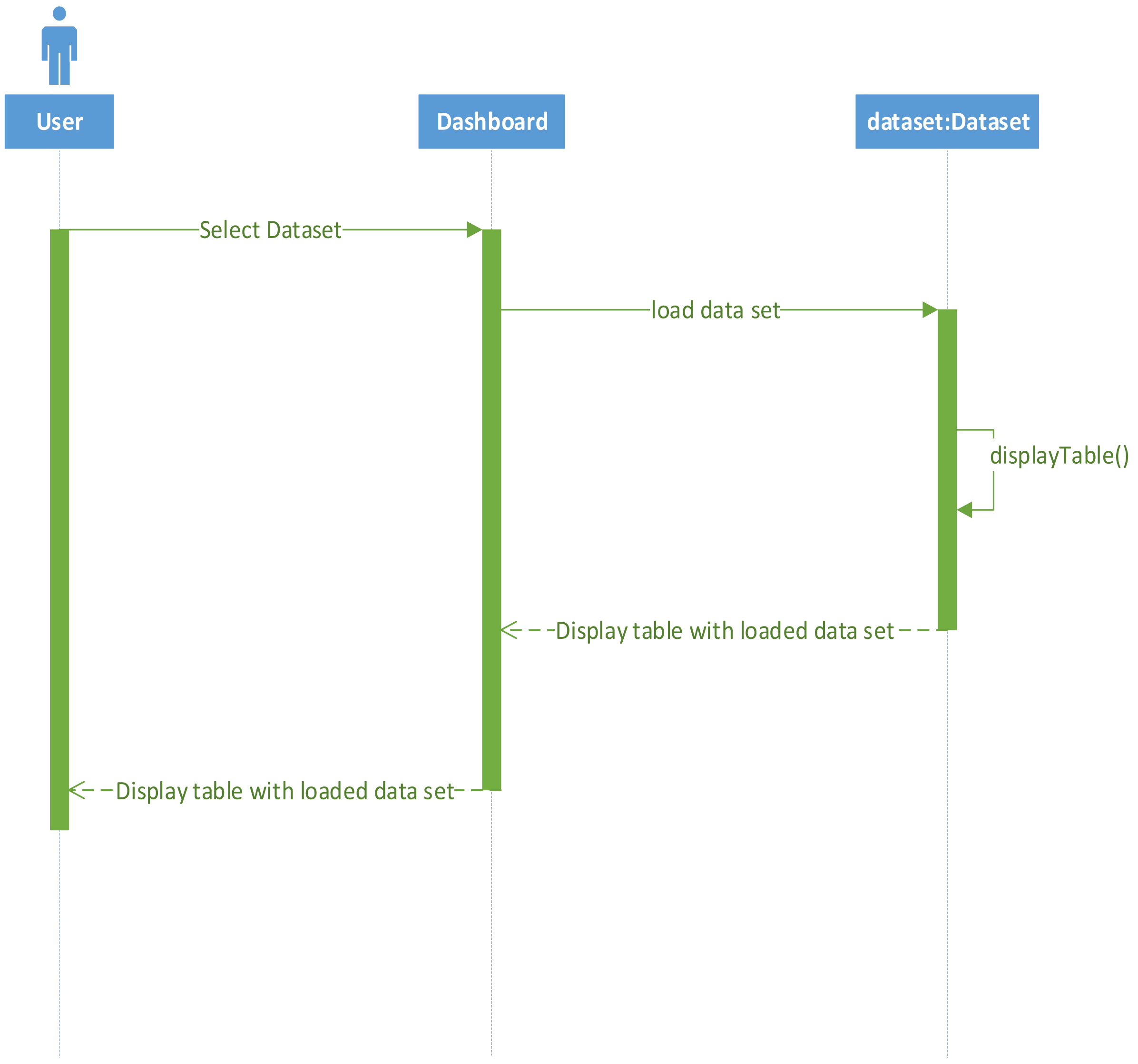

user
system
select 1 dataset from 5 dataset
n=5
datasets
1 dataset displayed successfully
select columns from selected dataset
selected columns displayed successfully
add filtering to selected dataset
filtered table displayed

iv) Sequence diagram for selecting and displaying charts


user
n=5
datasets
select 1 dataset from 5 dataset
1 dataset displayed successfully
select columns from selected dataset
selected columns displayed successfully
add filtering to selected dataset
filtered table displayed
select charts to be plotted
selected chart plottedd successfully

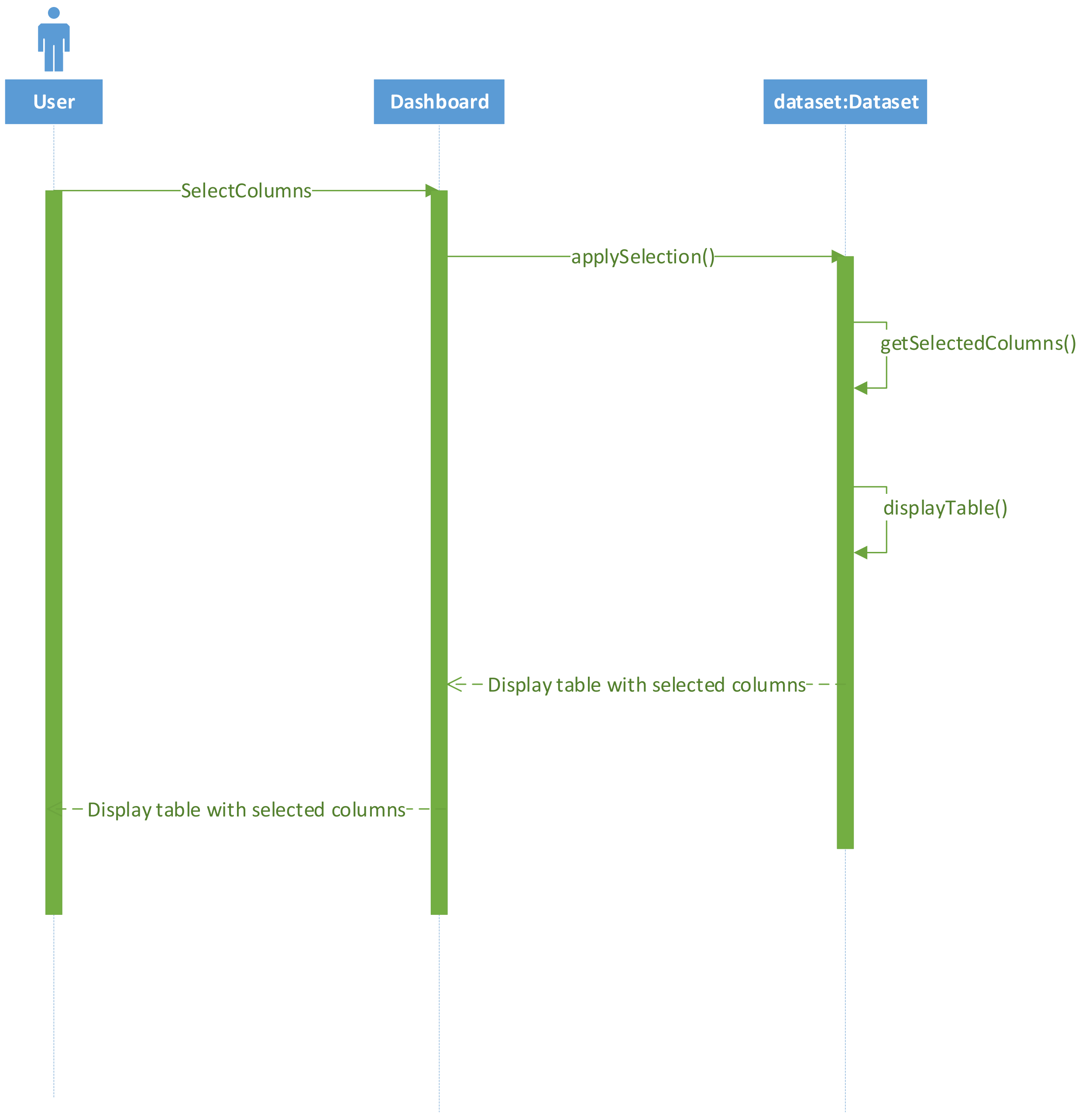
5.2 Sequence Interaction Diagrams

i) Sequence Diagram to select, load dataset and display in tabular format



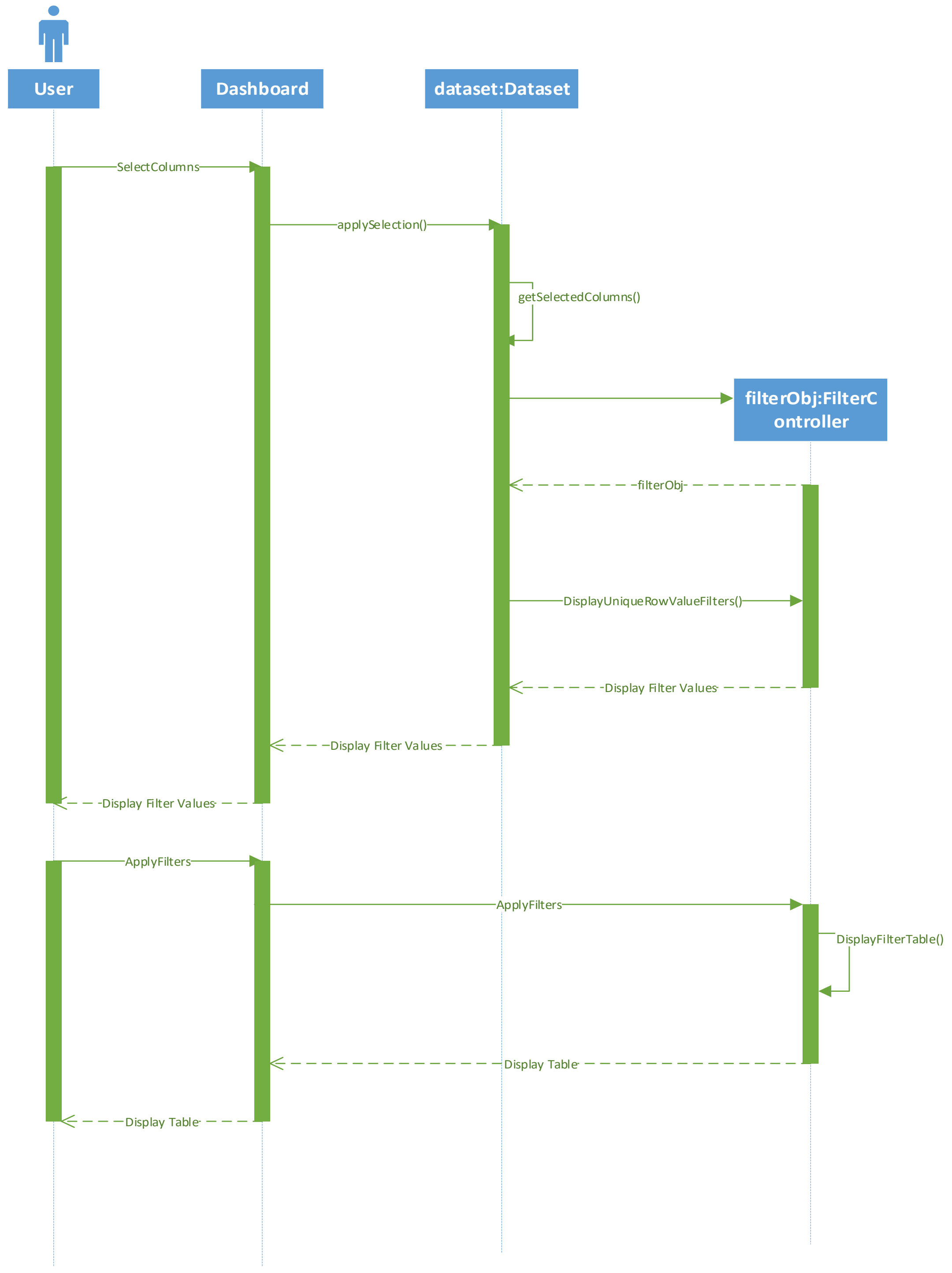
Sequence diagram to select, load dataset and display in tabular format

ii) Sequence diagram to select required columns from dataset



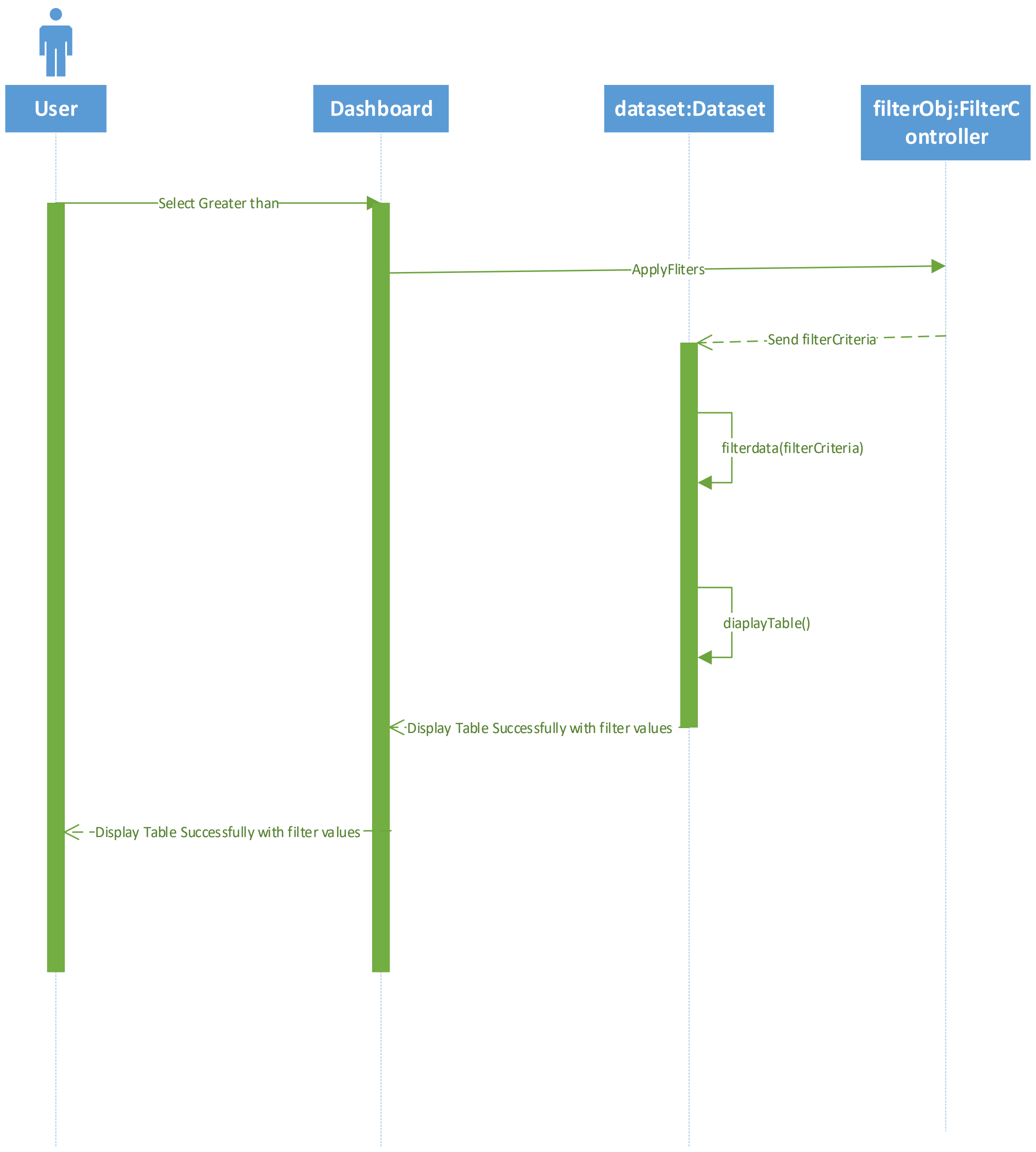
Sequence diagram to select required columns from dataset

iii) Sequence diagram to filter unique row values



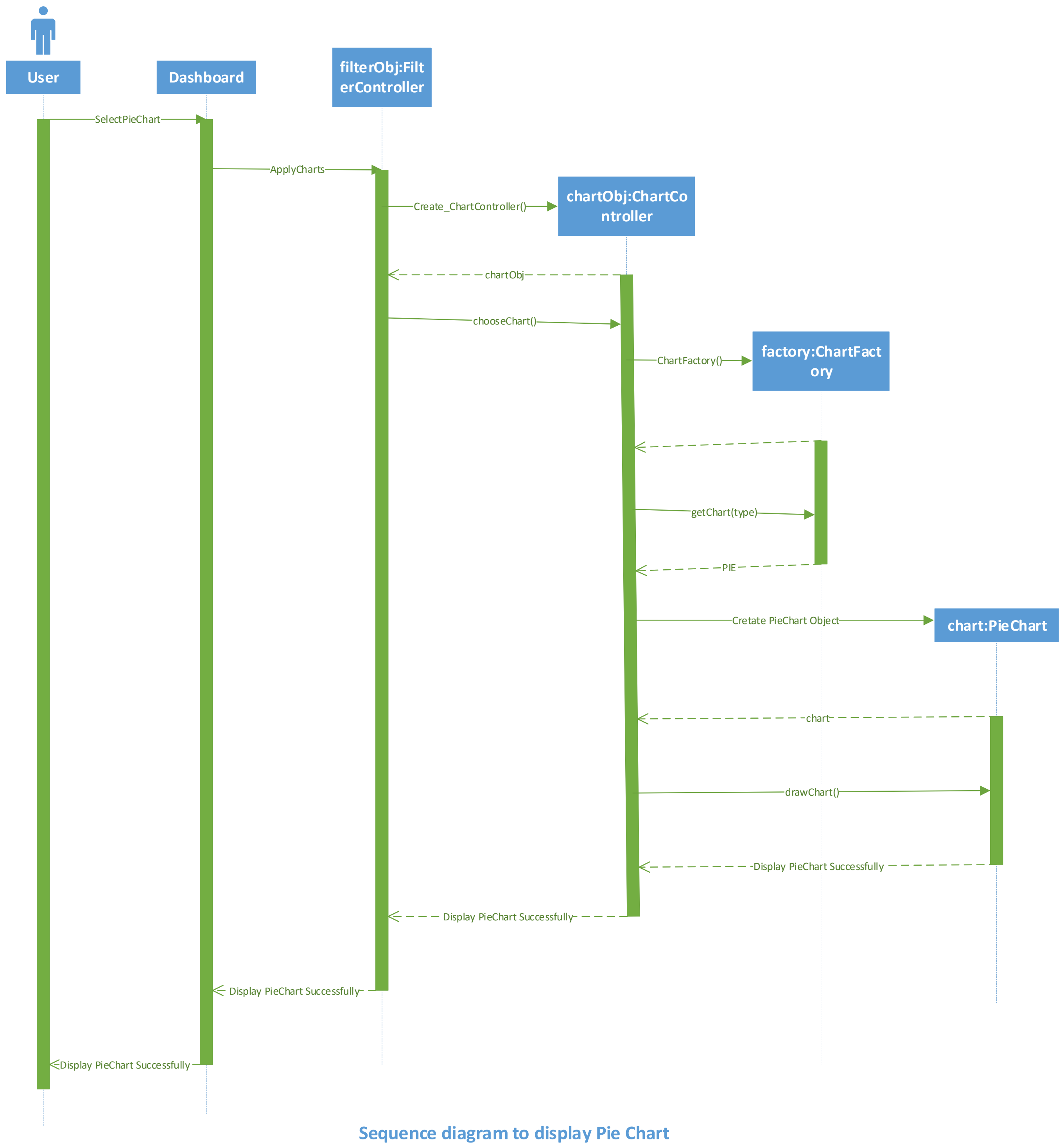
Sequence diagram to filter unique row values

iv) Sequence diagram to filter values with greater than filter criteria


User
Dashboard
dataset:Dataset
filterObj:FilterController
Select Greater than
ApplyFliters
Send filterCriteria
filterdata(filterCriteria)
diaplayTable()
Display Table Successfully with filter values
Display Table Successfully with filter values

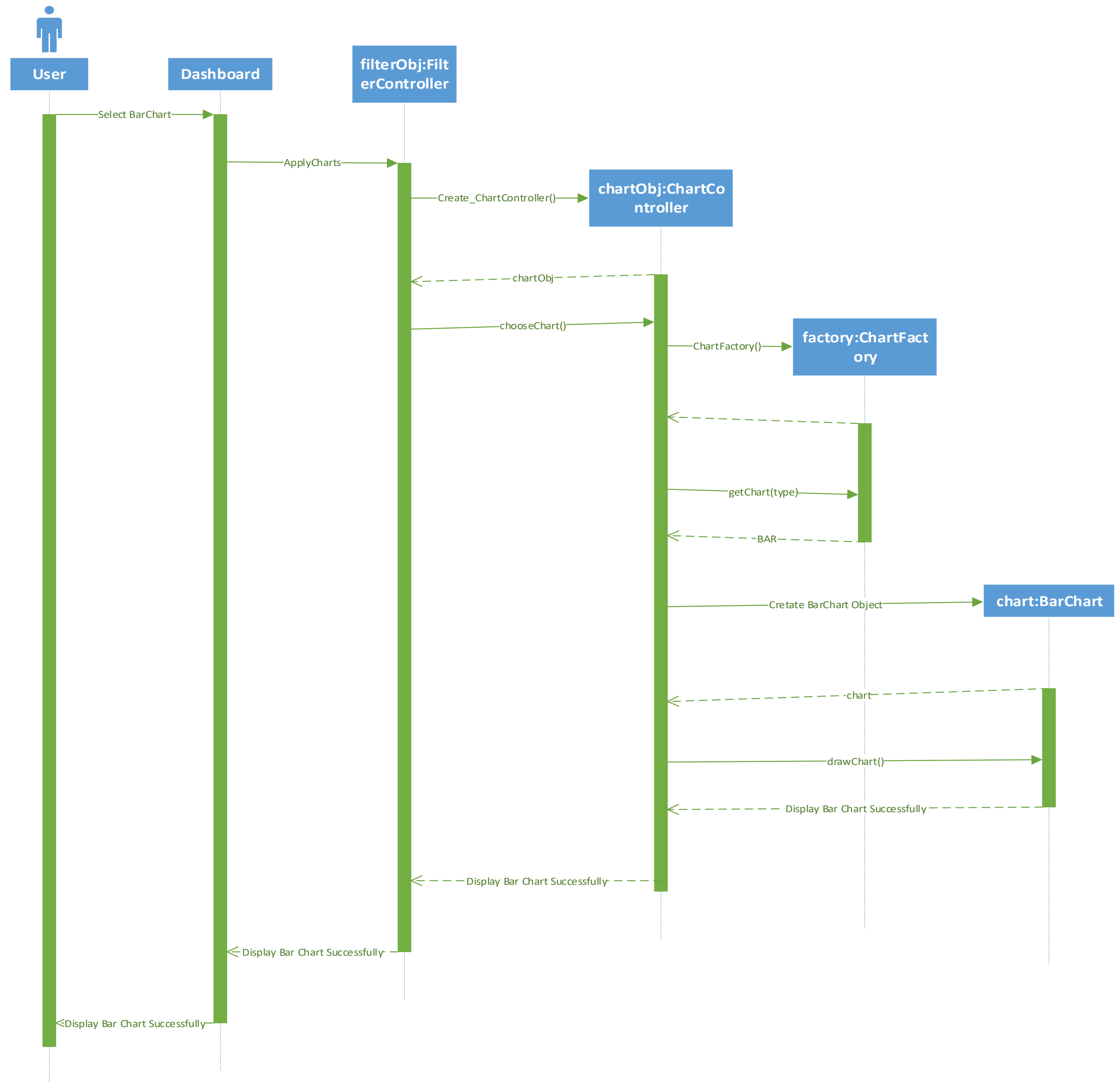
Sequence diagram to display filter values with greater than filter criteria

v) Sequence diagram to display Pie Chart



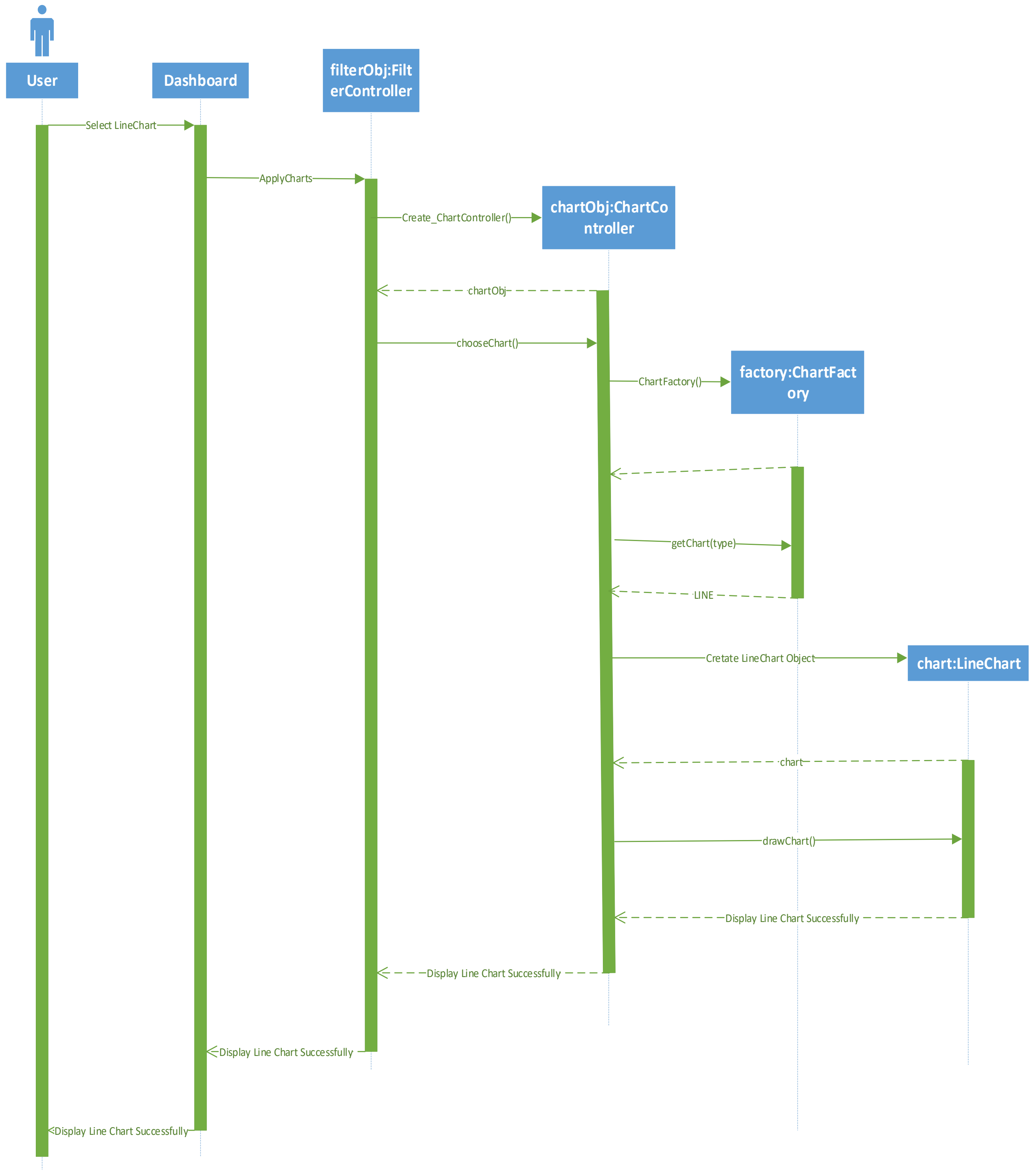
Sequence diagram to display Pie Chart

vi) Sequence diagram to display Bar Chart



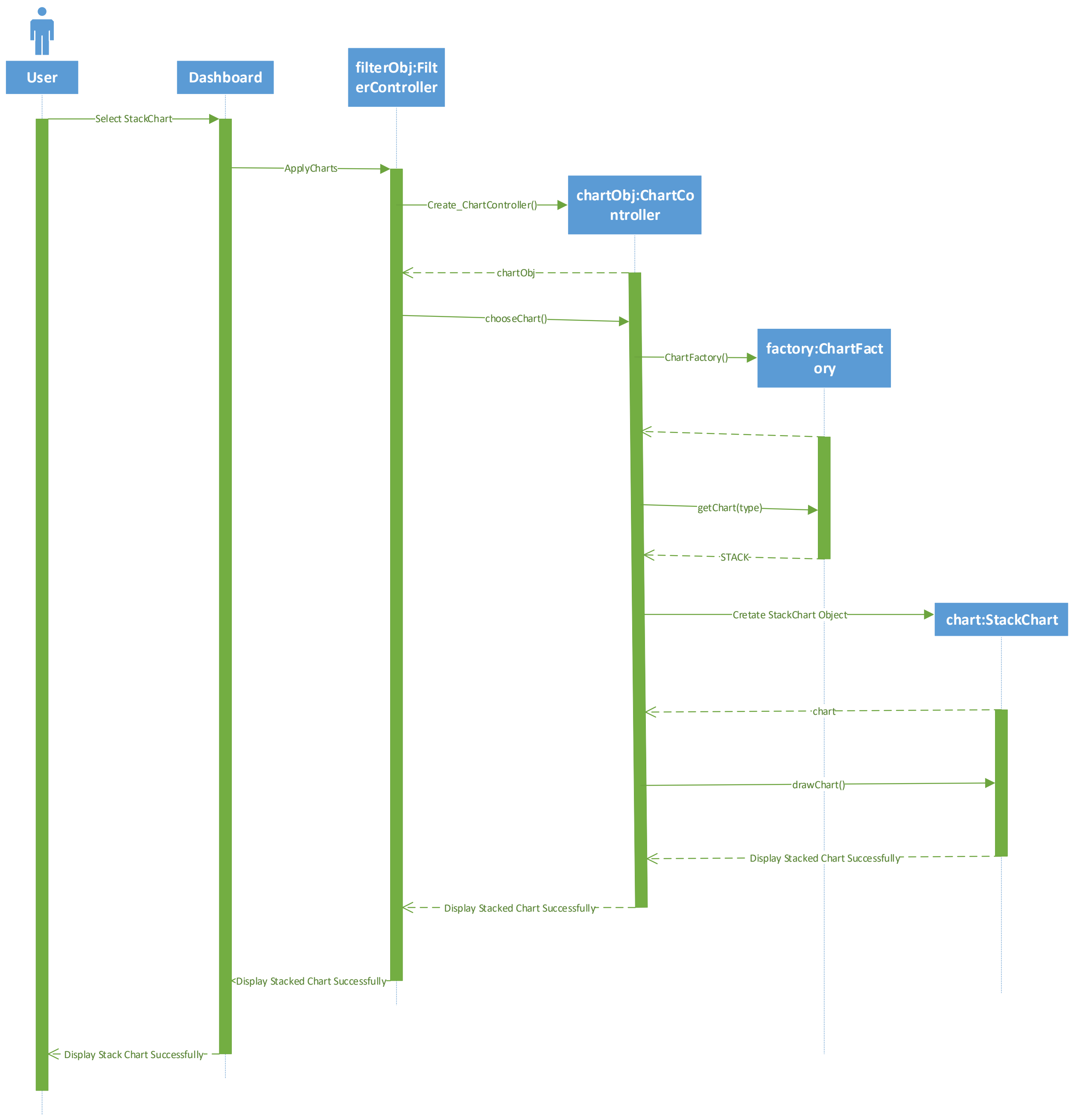
Sequence diagram to display Bar Chart

vii) Sequence diagram to display Line Chart



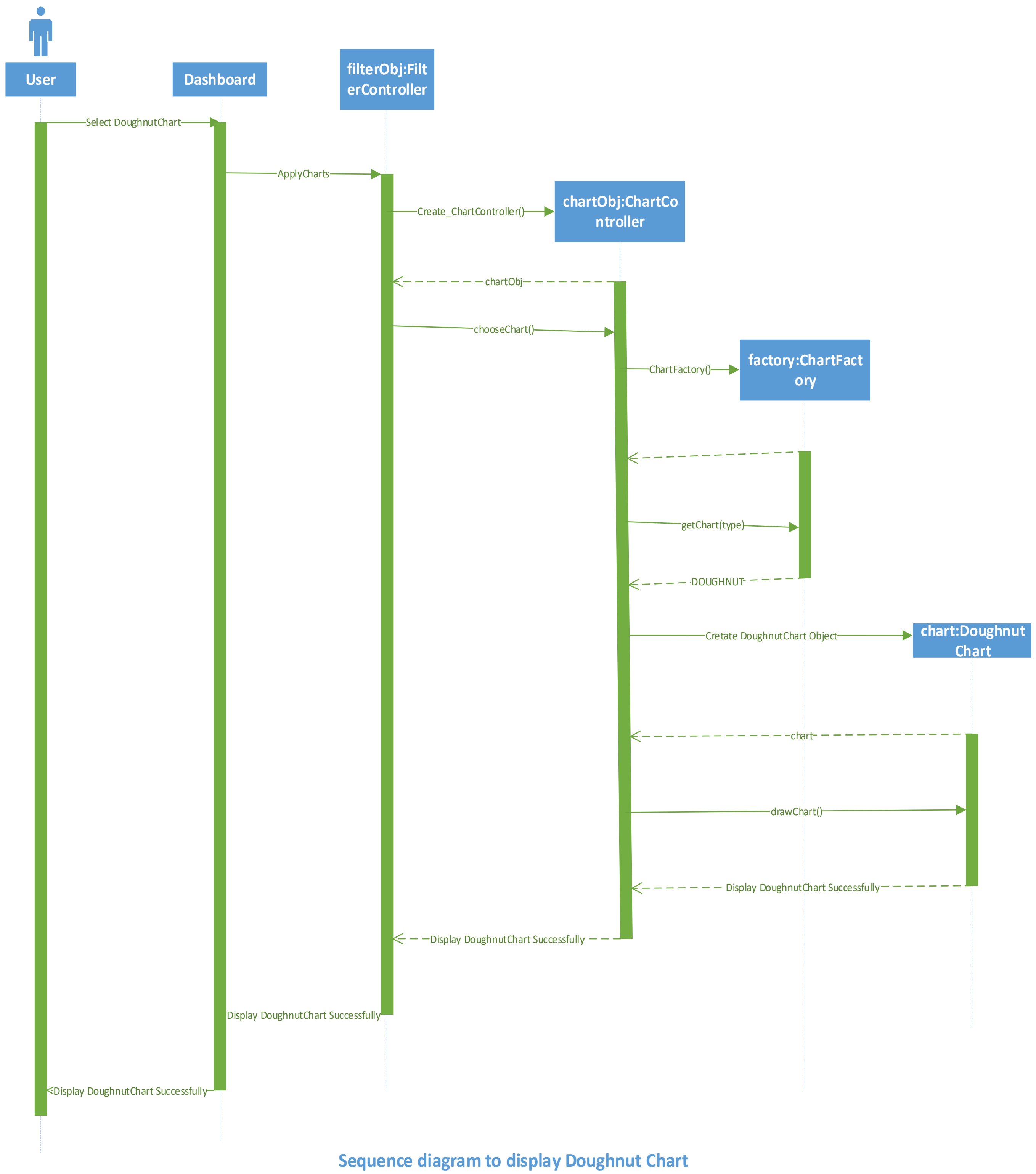
Sequence diagram to display Line Chart

viii) Sequence diagram to display Stacked Chart



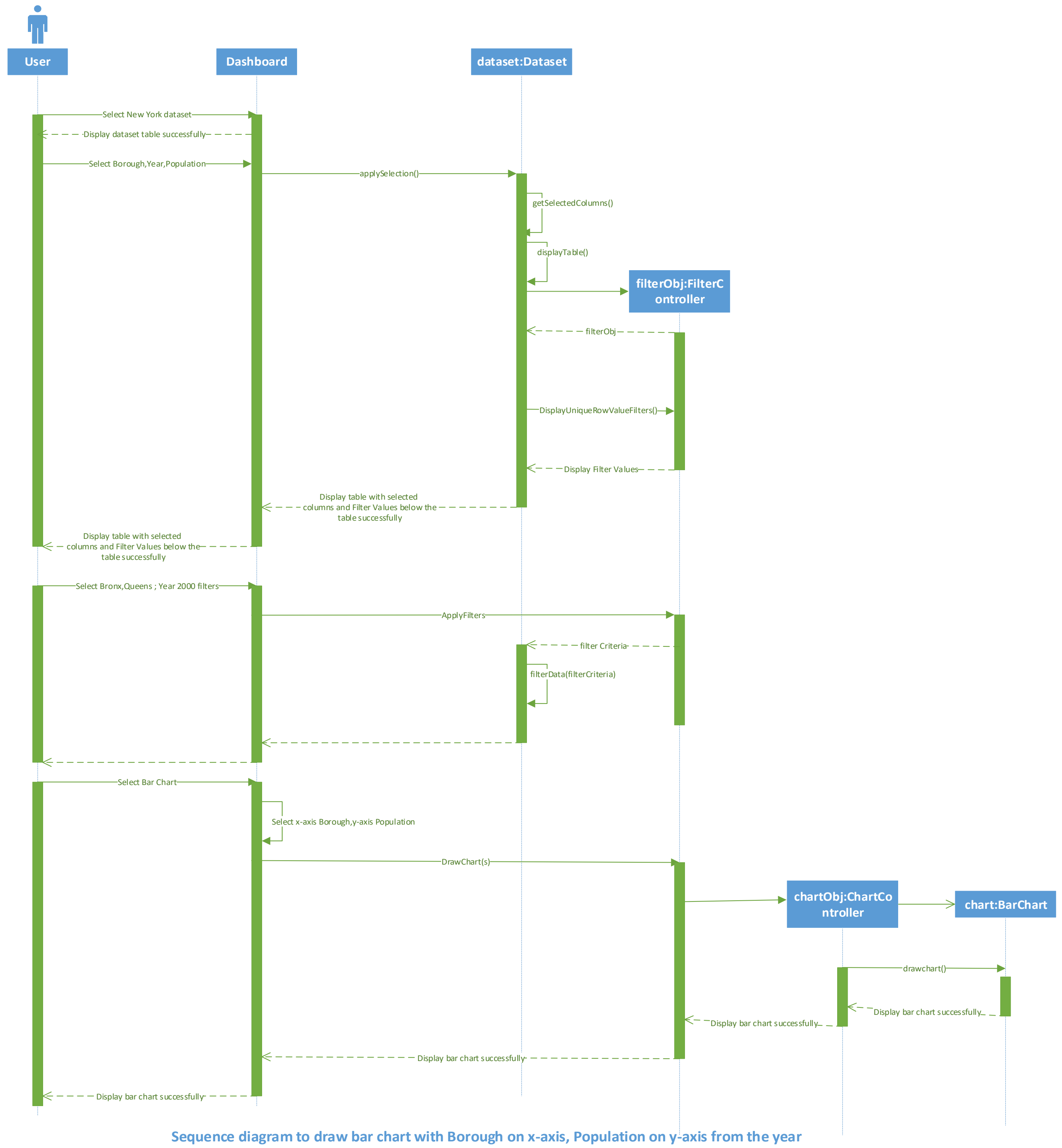
Sequence diagram to display Stacked Chart

ix) Sequence diagram to display Doughnut Chart



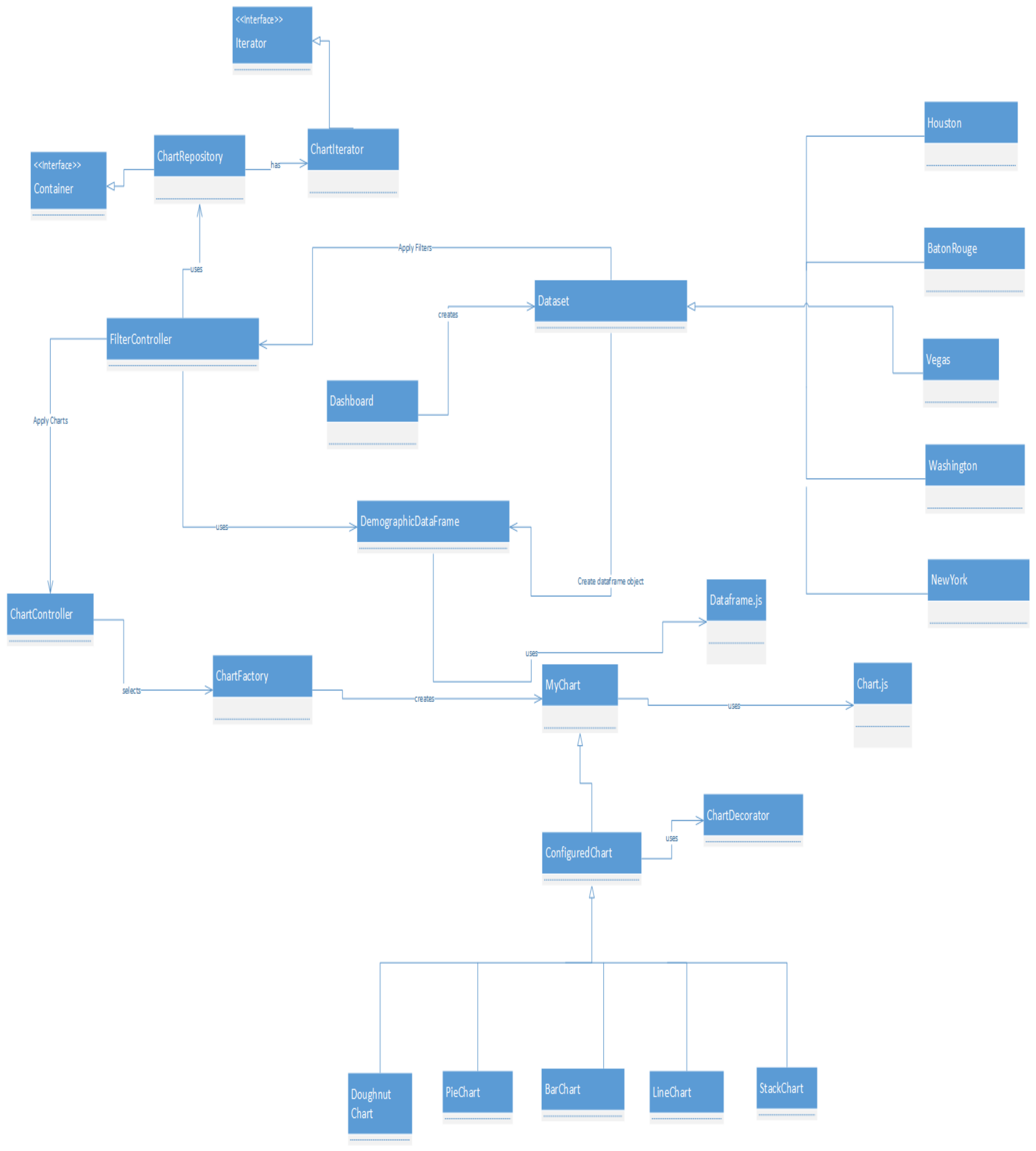
Sequence diagram to display Doughnut Chart

x) Sequence diagram to draw bar chart with Borough on x-axis, sum of Population on y-axis from the year 2000 in New York Census

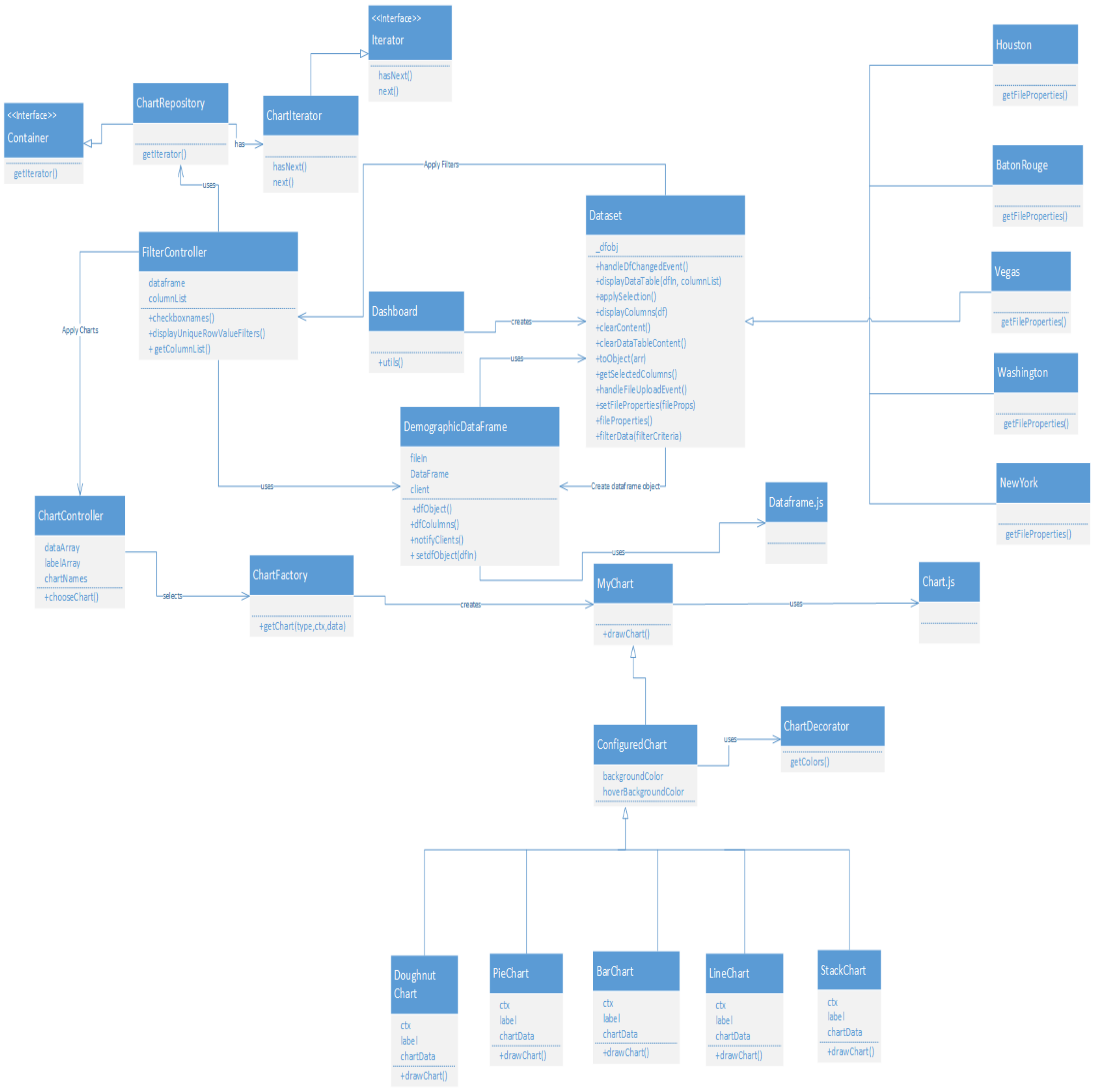

User
Dashboard
dataset:Dataset
Select New York dataset
Display dataset table successfully
Select Borough,Year,Population
applySelection()
getSelectedColumns()
displayTable()
filterObj:FilterController
filterObj
DisplayUniqueRowValueFilters()
Display Filter Values
Display table with selected columns and Filter Values below the table successfully
Display table with selected columns and Filter Values below the table successfully
Select Bronx,Queens ; Year 2000 filters
ApplyFilters
filter Criteria
filterData(filterCriteria)
Select Bar Chart
Select x-axis Borough,y-axis Population
DrawChart(s)
chartObj:ChartController
chart:BarChart
drawchart()
Display bar chart successfully
Display bar chart successfully
Display bar chart successfully
Display bar chart successfully

Sequence diagram to draw bar chart with Borough on x-axis, Population on y-axis from the year 2000 in New York Census

6. Domain Model Class Diagram:



7. Design Model Class Diagram:



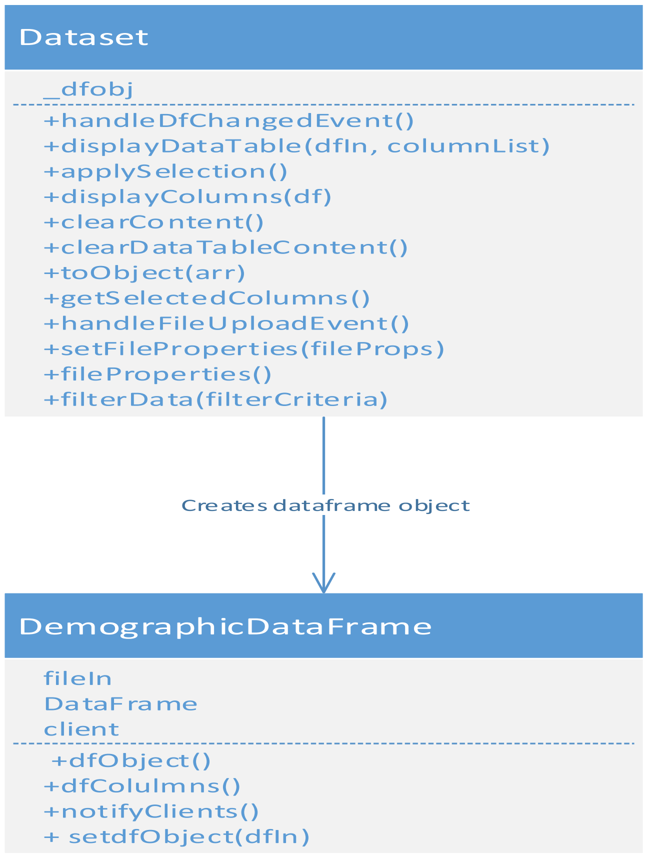
8. Design Patterns documentation

i. Singleton Design Pattern
ii. Factory Method Design Pattern
iii. Observer Design Pattern
iv. Iterator Design Pattern
v. Decorator Design Pattern

i. Singleton Design Pattern:

Singleton pattern ensures a class has only one instance and provides a global point of access to it. Design pattern comes under creational pattern

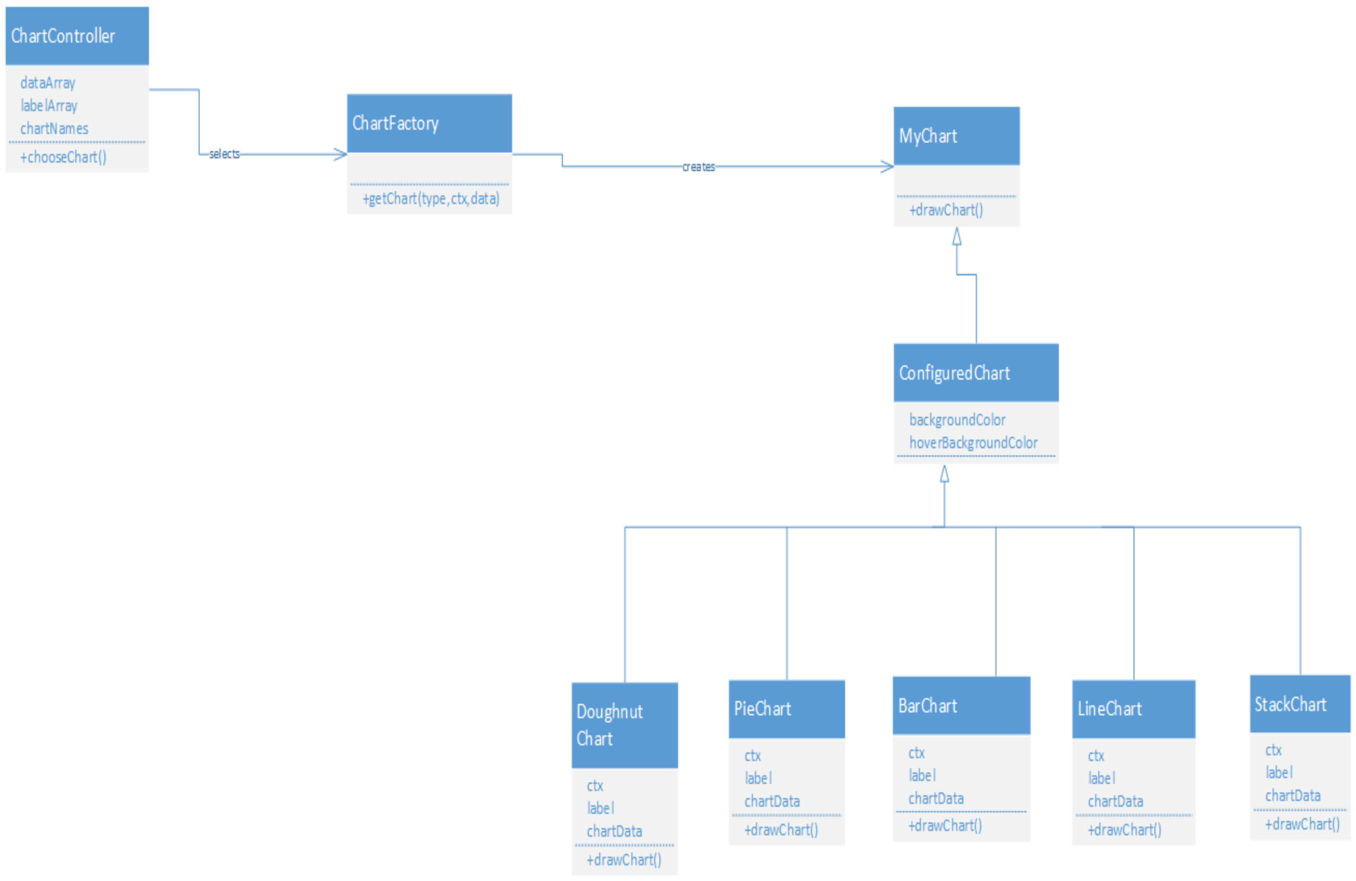
- DemographicsDataframe acts as singleton class
- dfObject() is the static method which returns DemographicsDataframe object
- Filtercontroller class uses DemographicsDataframe class to get df object



ii. Factory Method Pattern:

Factory Method pattern is one of the most widely used patterns. This comes under creational patterns as this pattern provides one of the best ways to create an object. In Factory Method pattern, we create an object without exposing the creation logic to the client and refer to newly created object using a common interface.

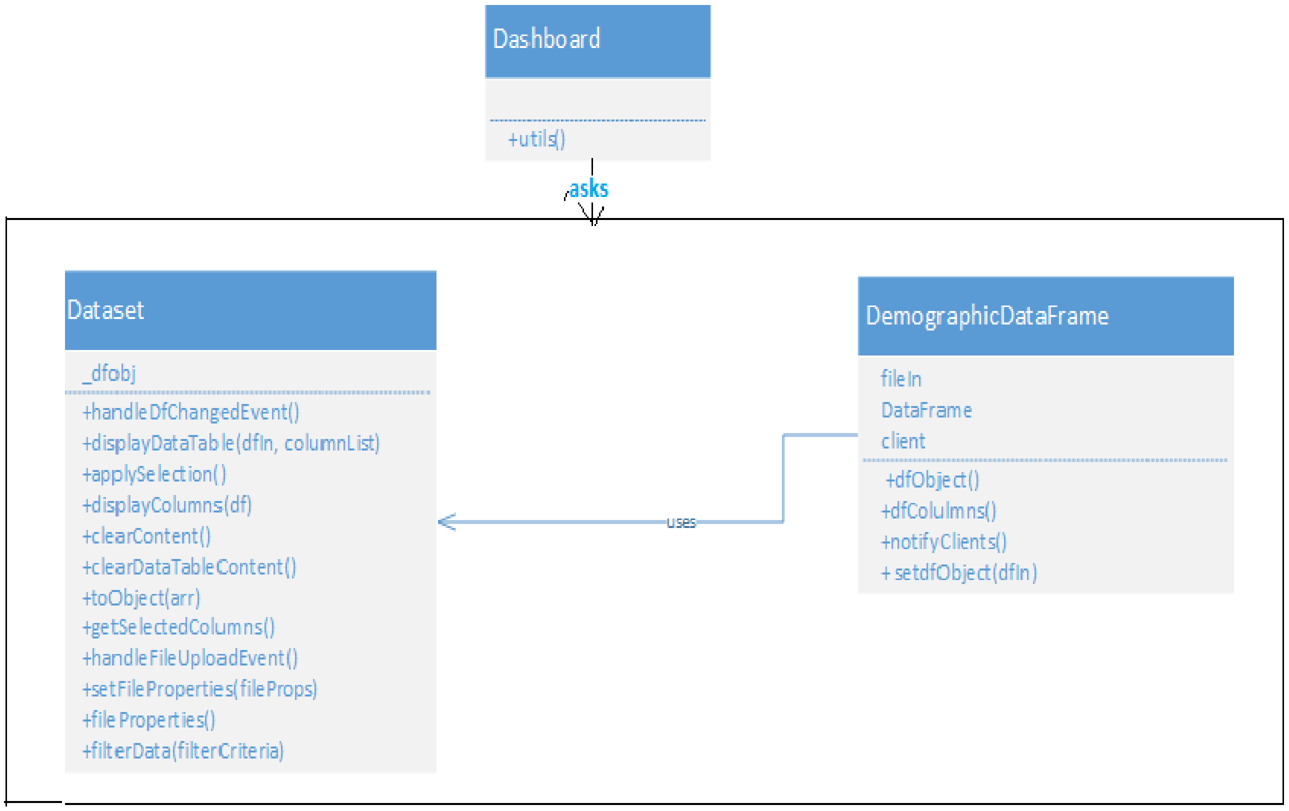
- Class “ChartFactory” is responsible for creating different **types** of chart classes.
- ”**ChartFactory**” object is created in “**ChartContoller**” class. “**MyChart**” class is defined as an abstract class and depends on the “ChartFactory” class. Bar, Pie, Stack, Line Doughnut, acts as concrete classes for MyChart class.
- ChartFactory has a method called “getChart (String type,ctx,data)” which accepts String as a parameter, object will be created based on the String value passed to this method. For instance, **getChart**(“PIE”) creates an object of class “Pie” which is a sub class of “MyChart” class. **drawChart()** method is used to display the chart of that class. Likewise, we can pass different String values as parameter to getChart () Method and create objects of different Chart classes.



iii. Observer Pattern:

Observer pattern is used when there is one-to-many relationship between objects such as if one object is modified, its dependent objects are to be notified automatically. Observer pattern falls under behavioral pattern category.

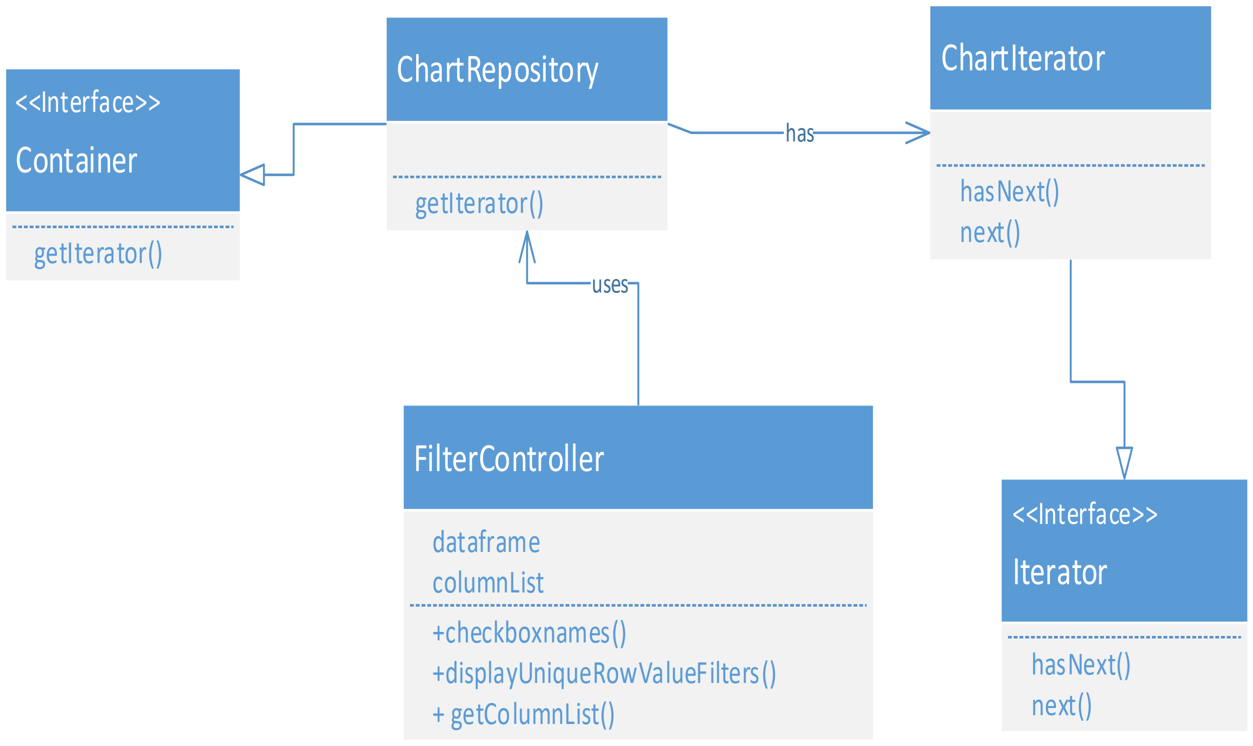
- Observer pattern uses three actor classes. Subject, Observer and Client. We have created a class called **Dataset(Observer)** and another class called **DemographicsDataframe (Subject)**. Class DemographicsDataframe has a method “**notifyClients()**” which will be invoked in case of any changes to dataframe object. This in turn calls handleDfChangedEvent() method on Dataset class. handleDfChangedEvent() then calls dfObject() object on DemographicsDataframe to get (modified) dataframe object.
- **Dashboard** class will use Subject and object class to show observer pattern in action.



iv. Iterator Design Pattern:

Iterator pattern allows traversal of the elements of an aggregate without exposing the underlying implementation. Iterator pattern falls under behavioral pattern category.

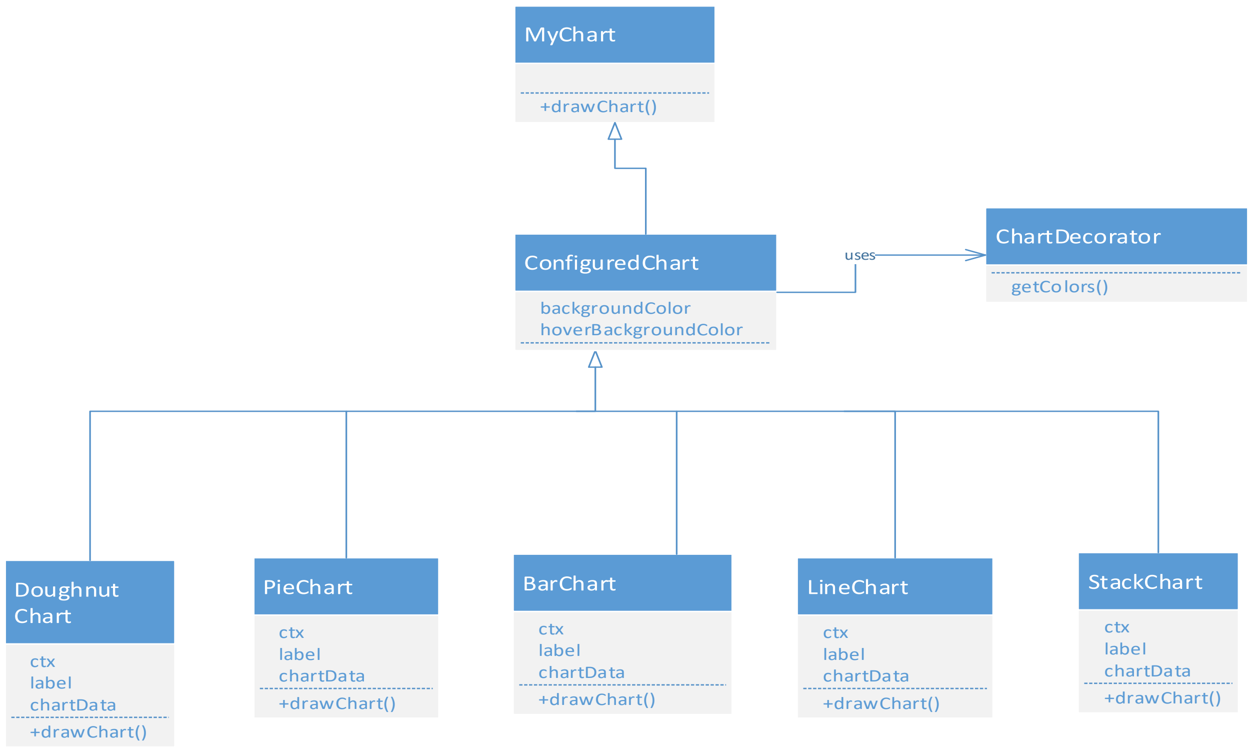
- **filterContoller** class uses **ChartRepository**, which acts as a concrete class to implement the Chart names stored as a collection in ChartRepository. Chart names are used to display the names of checkboxes.
- ChartRepository implements the Container interface. **getIterator()** method in ChartRepository returns the object of ChartIterator which is stored in "iter". It helps in iterating through the array of chart names.
- **ChartIterator** implements the **Iterator** interface. The iterator interface provides the interface that all the iterators must implement and a set of methods for traversing over elements of a collection.
- **hasNext()** method in the ChartIterator gives the information if there are more elements to iterate through. **next()** method returns the next object in collection.



v. Decorator Design Pattern

Decorator pattern attaches additional responsibilities to an object dynamically , i.e it allows to add new functionality to an existing object without altering its structure. This pattern comes under structural design pattern.

Each chart object is inherited from MyChart.js and ConfiguredChart.js, so each chart provides drawChart() function for displaying chart on browser. ChartDecorator.js class add getColors() function to existing ConfiguredChart object to display graph with different colors. Since ChartDecorator.js adding functionality to ConfiguredChart.js object, it is implementing Decorator pattern.



9. DataSet References:

https://catalog.data.gov/dataset/census-demographics

https://catalog.data.gov/dataset/census-tracts-2010-cd25c

http://geo.wa.gov/datasets/1502ded791fe408d8bc031ac9f63a361_0

http://cohgis-mycity.opendata.arcgis.com/datasets/census-2010-tracts

https://opendata.lasvegasnevada.gov/Planning/Census-Data-by-Tract/ddfn-86fa/data