**Scope of System:**

SortingApp system performs sorting operations and history.

**UI Layer:**

In UI layer, user can launch application by using url: <http://localhost:8080/SortingApp/Sorting.html>

User can enter random values by using the Add and Remove button operations.

When user clicks on Sort button, system displays the output in grid format with output, execution time and interchanged positions.

User can view history of operations performed by the user.

**Controller Layer**

In controller Layer, Rest calls captured from UI layer using DTO’s and sent to the business layer.

SortingApp System contains two rest calls for sorting and history purpose.

url1:http://localhost:8080/SortingApp/sorting

method :POST

url2:http://localhost:8080/SortingApp/sortinghistory

method :GET

**Business Layer**

In business layer, capture the request from controller and process the business logic and send the data to the DAO later to persist to the postgres database.

SortingApp system contains custom sorting logic, calculated execution time for process sorting operation and get soring history.

**DAO Layer**

In DAO layer, data persist into postgres database.

SortingApp System persists sorted results into sort table

**User interface (UI) files**

Sorting.html

app.js

sortingController.js

sortingService.js

**Configuration Files**

Configuration Files

SortingConfig.java

application.properties

**Controller File**

SortingController.java

**Business Layer Files**

SortingBusiness.java

SortingBusinessImpl.java

**Data Access Layer (dao) File**

SortingDao.java

**Entity and DTO**

SortingHistory.java

SoringList.java