

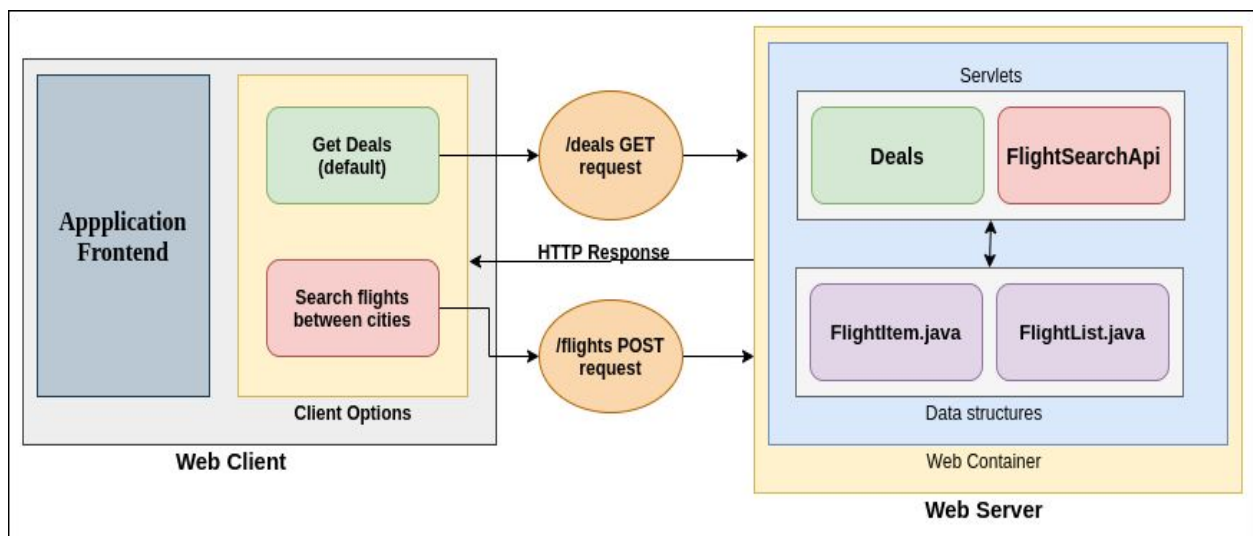
Problem Statement

Implement a web application for “Travel Thru Air” using servlets to support the following two use cases

1. A list of current special deals must appear on the home page. Each special deal must display the departure city, the arrival city, and the cost. These special deals are set up by the marketing department and change during the day, so it can't be static. Special deals are only good for a limited amount of time.
2. A user may search for flights, given a departure city, time and an arrival city. The results must display the departure city, the arrival city, the total cost, and how many legs the flight will have.

Design description

Structural design:



Design stack used:

1. Server

- a. Tomcat Server
- b. Servlet-api - module for implementing java servlets
- c. Json.simple-1.1.1 - module for using json objects

2. Client

- a. React-js Language
- b. React-Bootstrap - For frontend designing

Directory Structure:

<p>Wanderlust</p> <ul style="list-style-type: none"> — api <ul style="list-style-type: none"> — lib <ul style="list-style-type: none"> — json-simple-1.1.1.jar — servlet-api.jar — src <ul style="list-style-type: none"> — Deals.java — FlightItem.java — FlightList.java — FlightSearchApi.java — WEB-INF <ul style="list-style-type: none"> — classes <ul style="list-style-type: none"> — Ass3.class — Deals.class — Flight.class — FlightItem.class — FlightList.class — FlightSearchApi.class — SearchFlight.class — web.xml — ui <ul style="list-style-type: none"> — package.json — package-lock.json — public <ul style="list-style-type: none"> — index.html — manifest.json — robots.txt — src <ul style="list-style-type: none"> — App.css — App.js — index.css — index.js — Wanderlust.jpg 	<p>Api This directory contains backend JAVA-EE server code</p> <p>Deals.java Servlet to produce flight deals when requested by the client</p> <p>FlightSearch.java Servlet to generate flights as search results for client</p> <p>Web.xml Application deployment descriptor</p> <p>UI This directory contains client code</p> <p>App.js This is the main client code</p> <p>App.css Style sheets for the client</p>
--	---

Design Explanation:

Wanderlust is designed to display the current offers on flights to the users and provides options for searching flights between cities.

Web Server Files

1. **FlightItem.java** - This is used as a data structure to store the information of a single flight.
2. **FlightList.java** - This is used to store the list of flights available.
3. **Deals.java** - This is a servlet that responds to “/deals” GET requests coming to the server. This servlet responds with the information stored in FlightList object
4. **FlightSearchApi.java** - This is a servlet that responds to “/flights” POST requests coming to the server. This servlet accepts the source, destination and departure date from the request and searches the FlightList object to respond with the corresponding results.

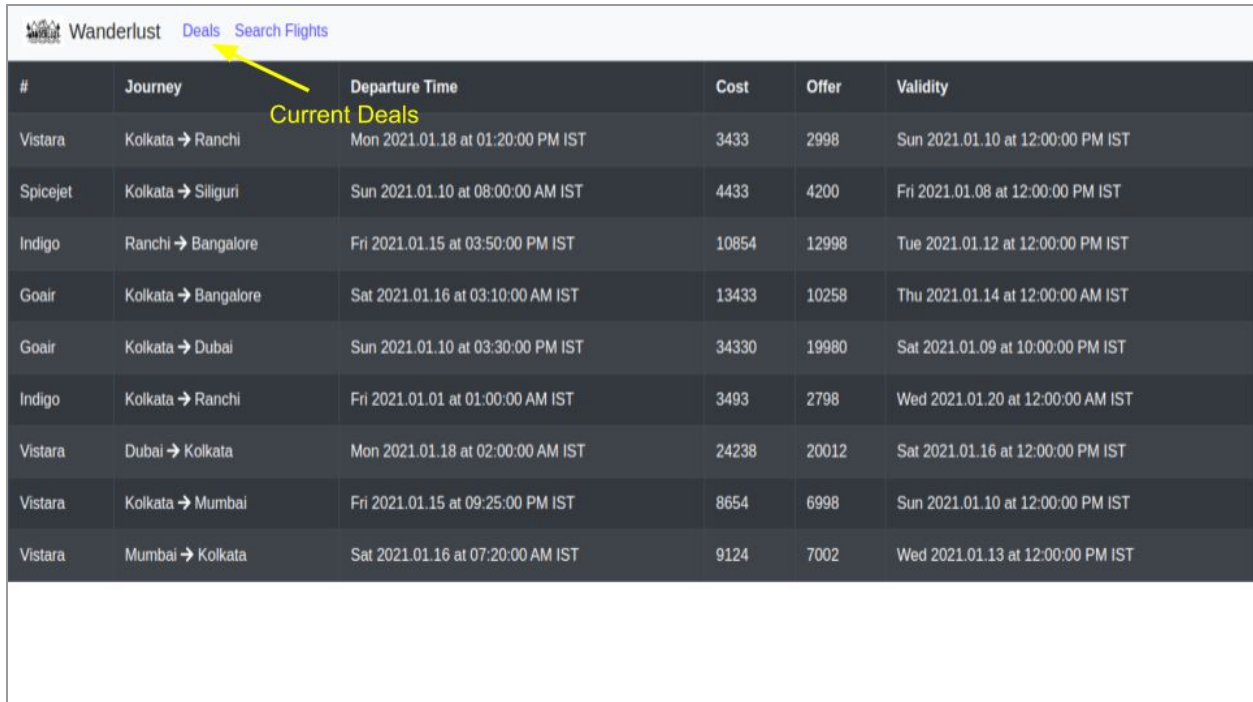
Web Client Actions

The app's client page has two options

1. **DEALS**(default) - This fetches the deals available from the web server using a “/deals” HTTP GET request.
2. **SearchFlights** - This asks the user to enter the source, the destination and the departure date of the journey. The recorded input is then sent to the server via “/flights” HTTP POST request.

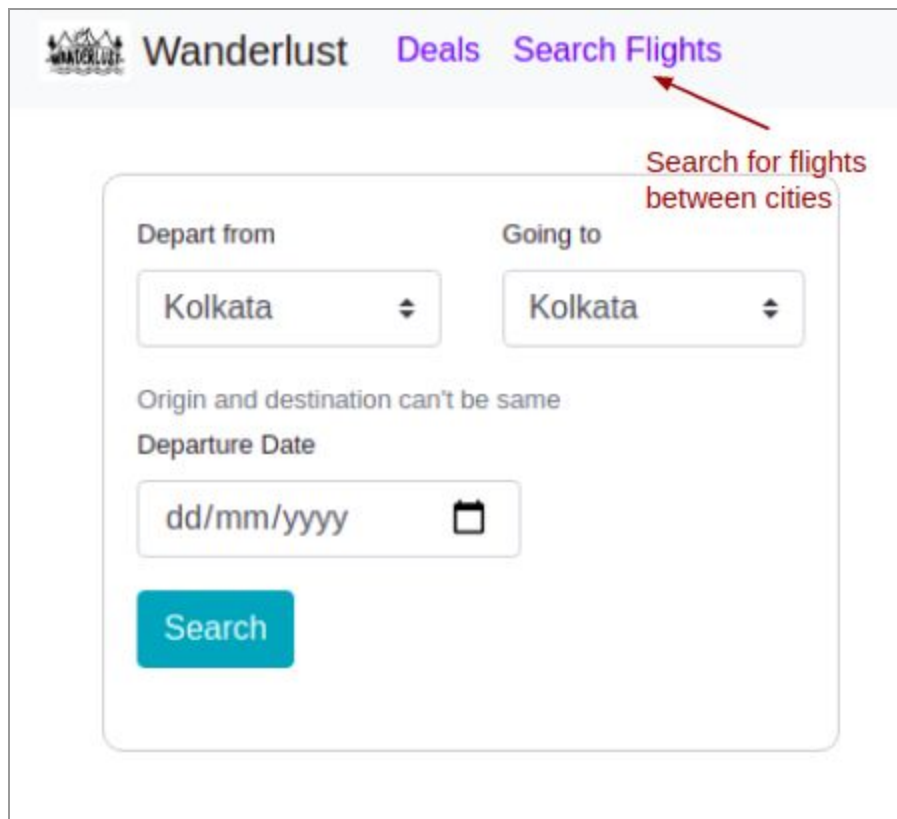
The response from the server is then processed and displayed to the user accordingly.

Sample Snapshots



#	Journey	Departure Time	Cost	Offer	Validity
Vistara	Kolkata → Ranchi	Mon 2021.01.18 at 01:20:00 PM IST	3433	2998	Sun 2021.01.10 at 12:00:00 PM IST
Spicejet	Kolkata → Siliguri	Sun 2021.01.10 at 08:00:00 AM IST	4433	4200	Fri 2021.01.08 at 12:00:00 PM IST
Indigo	Ranchi → Bangalore	Fri 2021.01.15 at 03:50:00 PM IST	10854	12998	Tue 2021.01.12 at 12:00:00 PM IST
Goair	Kolkata → Bangalore	Sat 2021.01.16 at 03:10:00 AM IST	13433	10258	Thu 2021.01.14 at 12:00:00 AM IST
Goair	Kolkata → Dubai	Sun 2021.01.10 at 03:30:00 PM IST	34330	19980	Sat 2021.01.09 at 10:00:00 PM IST
Indigo	Kolkata → Ranchi	Fri 2021.01.01 at 01:00:00 AM IST	3493	2798	Wed 2021.01.20 at 12:00:00 AM IST
Vistara	Dubai → Kolkata	Mon 2021.01.18 at 02:00:00 AM IST	24238	20012	Sat 2021.01.16 at 12:00:00 PM IST
Vistara	Kolkata → Mumbai	Fri 2021.01.15 at 09:25:00 PM IST	8654	6998	Sun 2021.01.10 at 12:00:00 PM IST
Vistara	Mumbai → Kolkata	Sat 2021.01.16 at 07:20:00 AM IST	9124	7002	Wed 2021.01.13 at 12:00:00 PM IST

Fig 1. Default page of the app showing available deals



The screenshot shows the Wanderlust website's 'Search Flights' page. The header includes the Wanderlust logo and navigation links for 'Deals' and 'Search Flights'. The search form contains two dropdown menus for 'Depart from' and 'Going to', both set to 'Kolkata'. Below these is an error message: 'Origin and destination can't be same'. The 'Departure Date' field is set to 'dd/mm/yyyy' with a calendar icon. A red arrow points to the 'Search Flights' link with the text 'Search for flights between cities'.

Wanderlust Deals Search Flights

Search for flights between cities

Depart from: Kolkata

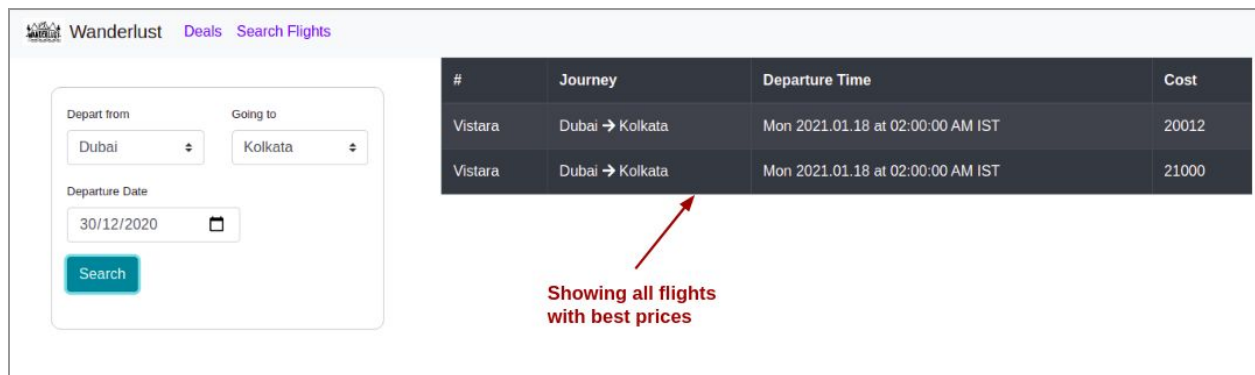
Going to: Kolkata

Origin and destination can't be same

Departure Date: dd/mm/yyyy

Search

Fig 2. Search Flights page with search form and error checking



The screenshot shows the Wanderlust website's 'Search Flights' page with search results. The search form on the left has 'Depart from' set to 'Dubai' and 'Going to' set to 'Kolkata'. The 'Departure Date' is set to '30/12/2020'. A red arrow points to the search results table with the text 'Showing all flights with best prices'.

Wanderlust Deals Search Flights

Search for flights between cities

Depart from: Dubai

Going to: Kolkata

Departure Date: 30/12/2020

Search

#	Journey	Departure Time	Cost
Vistara	Dubai → Kolkata	Mon 2021.01.18 at 02:00:00 AM IST	20012
Vistara	Dubai → Kolkata	Mon 2021.01.18 at 02:00:00 AM IST	21000

Showing all flights with best prices

Fig 3. Search Flight results from the server