Flight Ticket Booking

# Problem

Mr. Mark wants to implement Salesforce in his new business i.e. Flight Ticket Booking System. He has 70+ ticket booking centers across country. He wants to manage everything using salesforce like booking process, monthly reports, offers on flights, user management like Admin, CEO, Manager, Employee etc.

# Planning

As per Mark requirement, Salesforce is able to provide each and everything needed by him. There are different Edition of Salesforces are available for use. Lighting Professional would be best for him in initial stage of business late on can change plan anytime. It includes $ 75 USD/user/month (billed annually). Besides every functionality, it includes two version of Sandboxes I.e. Developer Sandbox, Developer Pro Sandbox. These would help production and development team can work simultaneously without affecting others work. Basic requirements of system would be finish by end of this month with integration and testing.

# System Analysis & Requirements

* **Lighting Professional Edition (Salesforce) -$75 USD/user/month (billed annually).**
* **Function Requirements**
* Flight search between source and destination.
* Listing all flights with sorting option like cheapest first, timing, travel time etc. and filter like non-stop, via some location.
* Booking of ticket. Generating ticket after payment received.
* Ticket Email to customer
* Manager can view reports and manage employees.
* Admin user with all privileges except customer sensitive information like Bank account details, Social security number etc.
* **Software and Tools**
* **Developer Console (Salesforce):**

The Developer Console is an integrated development environment with a collection of tools you can use to create, debug, and test applications in your Salesforce organization.

* **Force.com IDE (for Eclipse):**

The Force.com IDE is an integrated development environment for developing applications on the Force.com platform using Apex, Visualforce, and metadata components. Designed for developers and development teams, the IDE provides tools to accelerate Force.com application development. Using plugin for eclipse.

* **GIT for Version control:**

Git is a version control system for tracking changes in computer files and coordinating work on those files among multiple people. It is primarily used for source code management in software development, but it can be used to keep track of changes in any set of files.

* **Bitbucket.org (online Repository):**

Bitbucket is a web-based hosting service that is owned by Atlassian, used for source code and development projects that use either Mercurial (since launch) or Git (since October 2011) revision control systems. Bitbucket offers both commercial plans and free accounts.

* **Technical Knowledge and learning**
* **Visualforce Pages**

Visualforce is the component-based user interface framework for the Force.com platform. The framework includes a tag-based markup language, like HTML. Each Visualforce tag corresponds to a coarse or fine-grained user interface component, such as a section of a page, or a field.

* **Lightning Component & Aura**

The Lightning Component framework is a UI framework for developing dynamic web apps for mobile and desktop devices. It's a modern framework for building single-page applications engineered for growth.

* **Apex**

Apex is a proprietary language developed by Salesforce.com. It is a strongly typed, object oriented programming language that allows developers to execute flow and transaction control statements on the Force.com platform server in conjunction with calls to the Force.com API.

* **HTML**

Hypertext Markup Language, a standardized system for tagging text files to achieve font, color, graphic, and hyperlink effects on World Wide Web pages.

* **JavaScript**

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities

* **CCS3**

CSS3 has been split into "modules". It contains the "old CSS specification" (which has been split into smaller pieces). In addition, new modules are added.

* **SLDS Design**

The Lightning Design System provides accessible markup which will serve as a foundation for your application development. To build accessible components, you will need to follow the accessibility guidance for our interactive components, including keyboard behavior and the management of ARIA roles and properties

# System Design

This system basically contains three modules Admin, Employees, and Managers.

1. **Admin:**

* Create New Route
* Create New Flight
* Flight List
* Update Flight
* Add Flight Status
* Flight Status List
* Update Flight Status
* Add offers
* Manager Users
* Customize reports and send to higher Authorities.

1. **Employee:**

* Booking
* Booking History
* Cancellation
* Seat Availability
* Enquiry
* Flight Status
* Change Password

1. **Manager:**

* See status of user employees
* Information Admin to block any user.
* Able to see all data flight, booking, users who are working under him.

# Development

I have different my development and testing in several modules. Basic goal for development was to develop a Lightning one-page App which would help agent to book ticket.

Let’s discuss each module in detail.

1. **Creating Profile**:

I have created one more profile for testing purpose. This profile has view only permission. Agent with user ID: [guest@flight.com](mailto:guest@flight.com) has been tag to this profile. Using this profile user can view and book ticket.

1. **Created Custom domain:**

To run lightning app, I have created one custom domain using my developer account. i.e. https://pankajdwivedi-dev-ed.lightning.force.com

1. **Creating Object**:

There are three custom objects:

1. FlightDetail:

This object contains fields like Flight Name, Flight departure, Flight Arrival Details, fare, travel time, total seats.

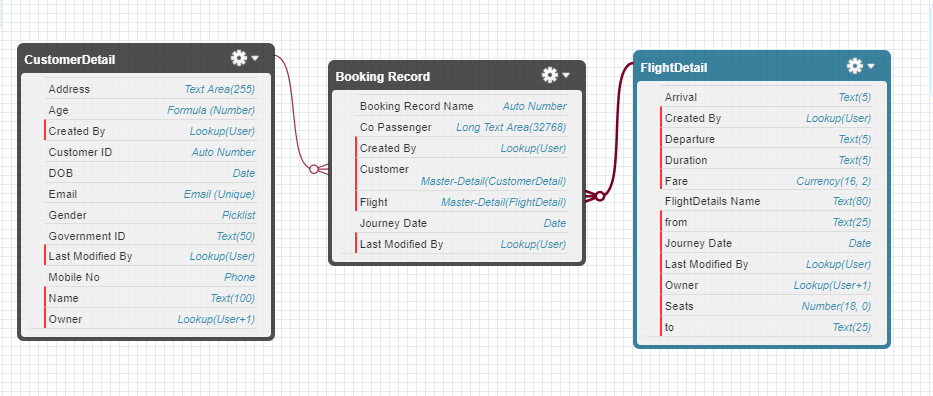
1. CustomerDetail:

This object contains fields like Customer Name, Customer Email, DOB, address, mobile number.

1. BookingDetail:

This object contains fields like date of booking, order number, flight reference, Customer reference

Following figure would describe all of three objects:



1. **Creating Lightning Component:**

There 5 lightning components.

1. Home.cmp

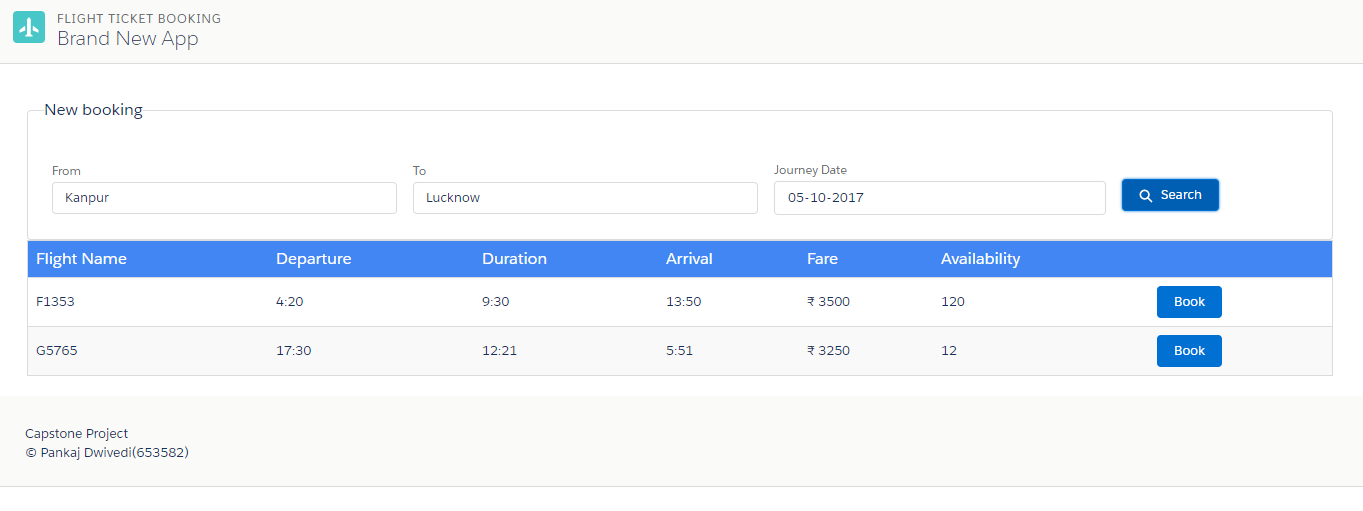
This component gets load once agent hint the URL. This means this is starting page for booking.

Home component have two sections first section enter search item details like source city, destination city and date of journey. Once search button clicked it call the controller method “c.getSearchFlights”. Component Controller written in JavaScript validated the data then call Apex method “c.searchFlight”.

Once data comes from server,it would displayed in tables

**Addition Features:**

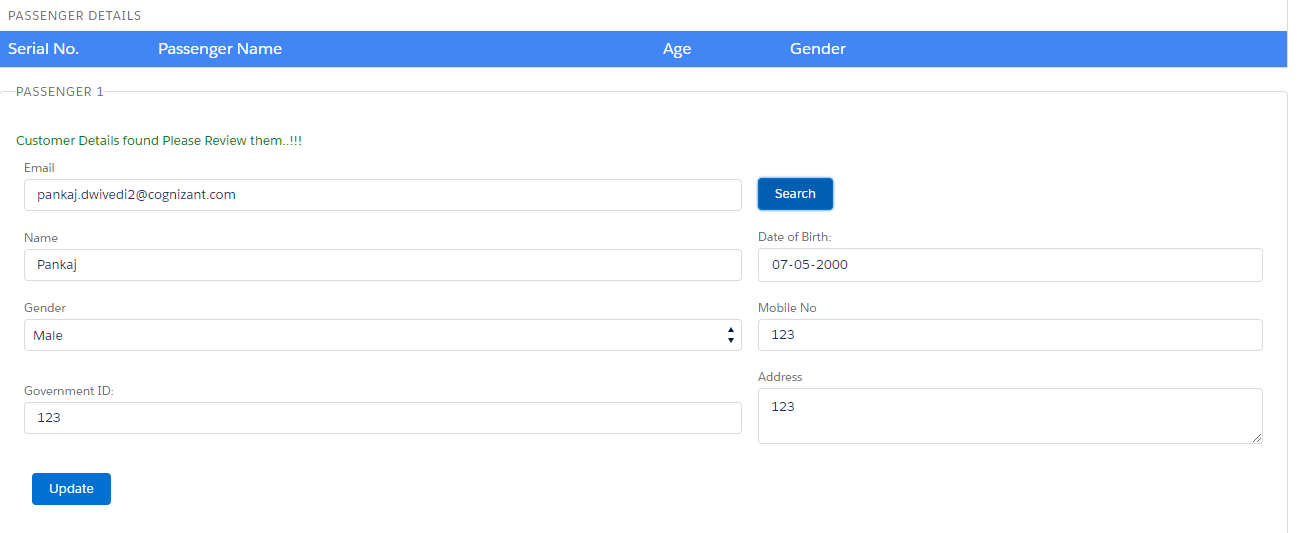
There auto suggestion for from and to city stored in our salesforce database.

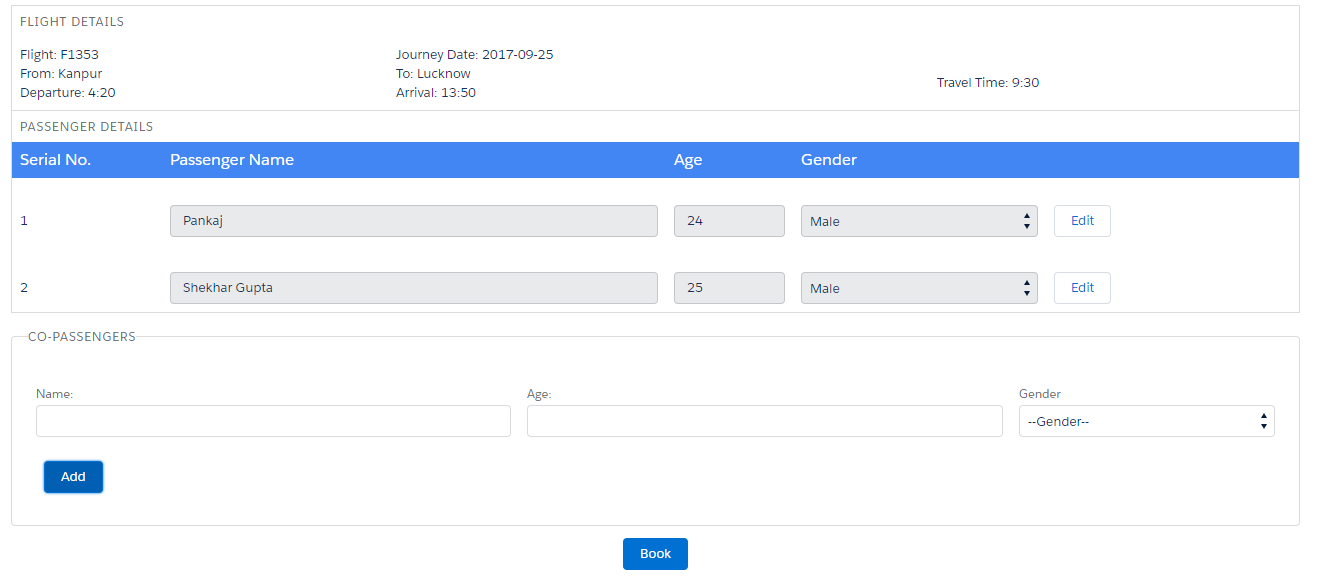


1. AddPassenger.cmp

In This Page Agent can search exiting customer by their email id. If existing user details would be added else agent can add him/her. As new customer.

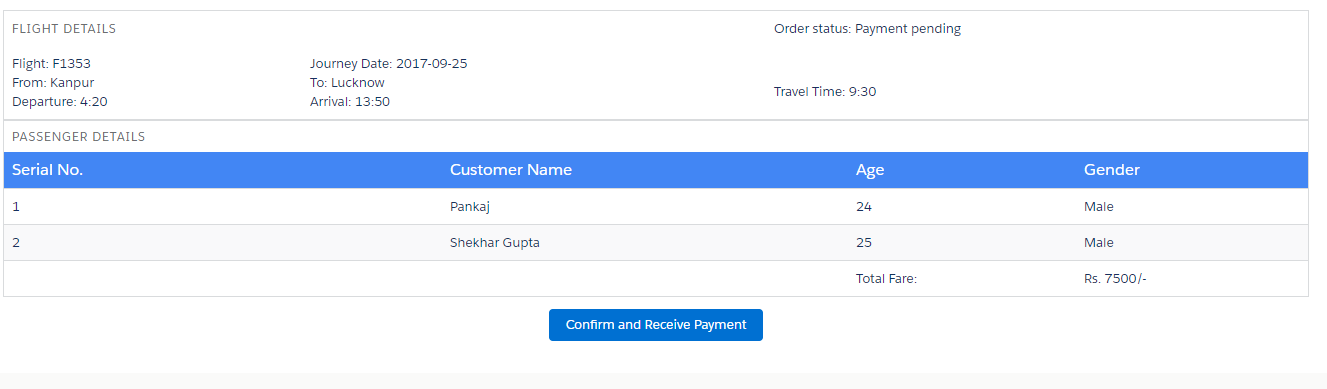
Agent can several Co-passengers too. After the click on Book.

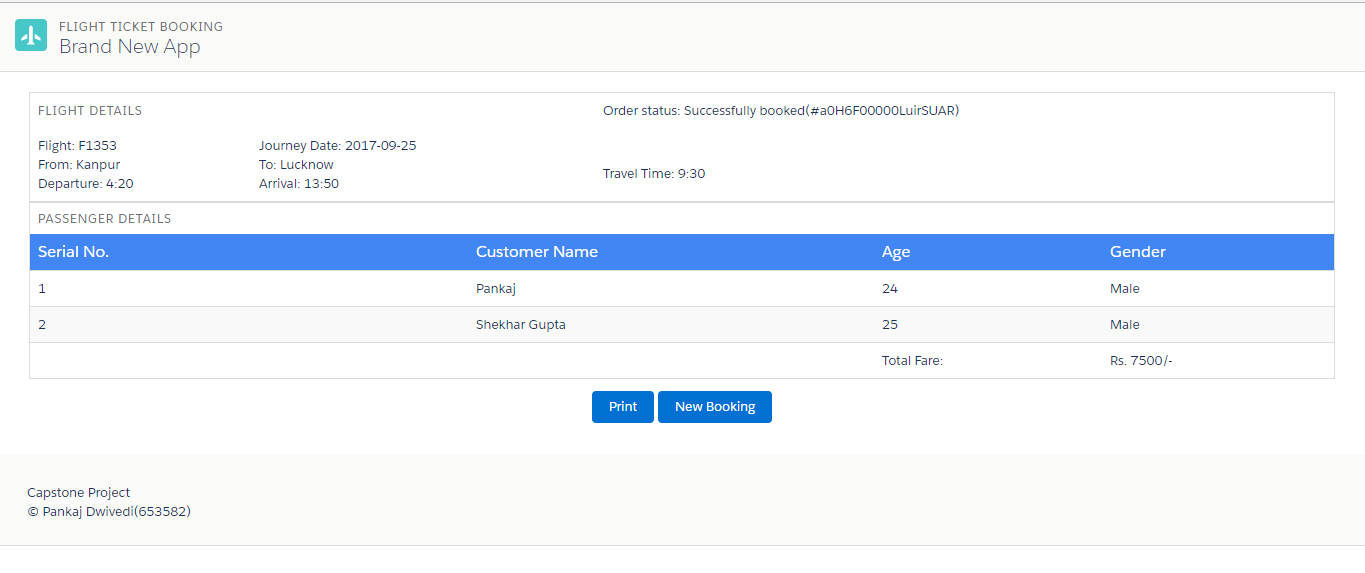


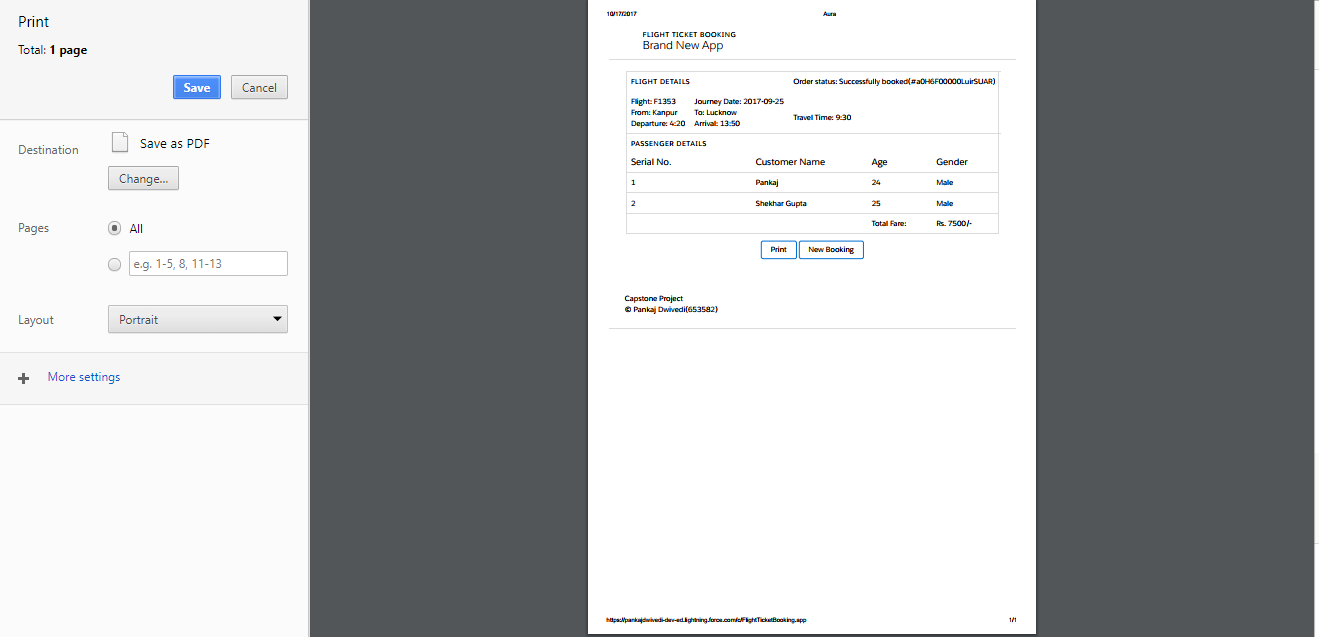


1. PaymentPage.cmp

In page Agent ask the customer to confirm the details enter by him. Once details are confirmed and Agent received the payment click on Book then apex method Book\_ticket is called. After successful database entry. It sends the email to customer and print button appear for printing the ticket.

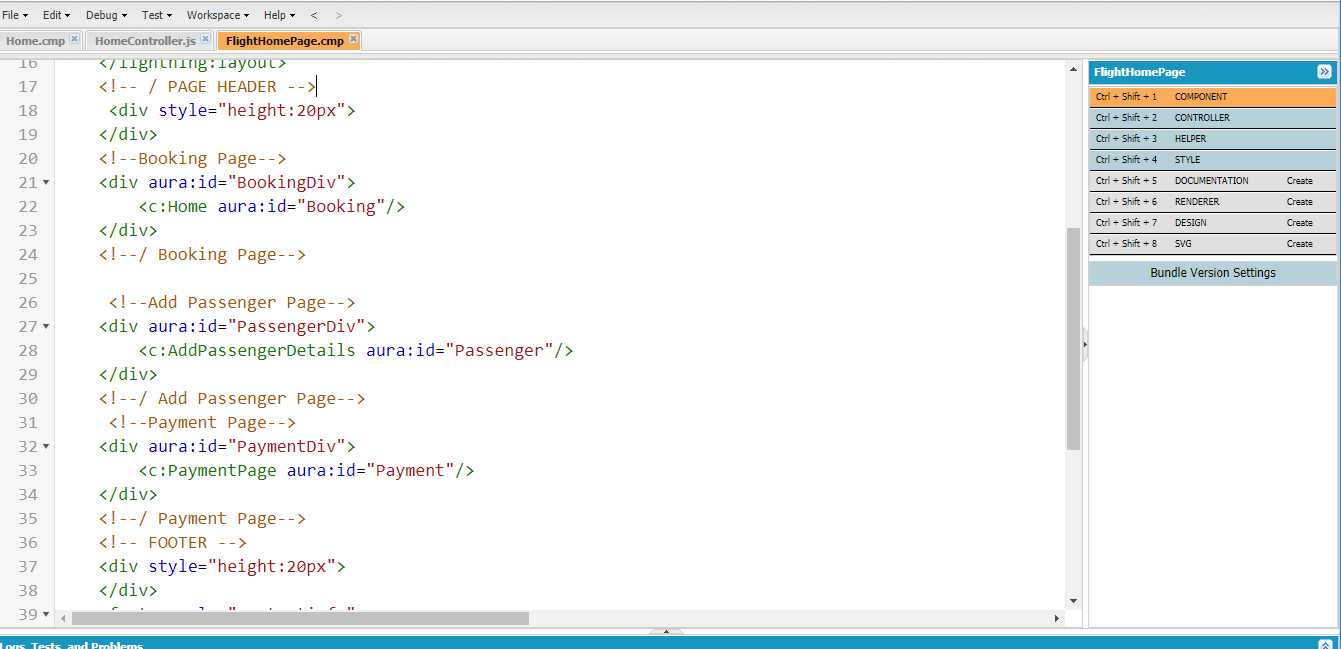






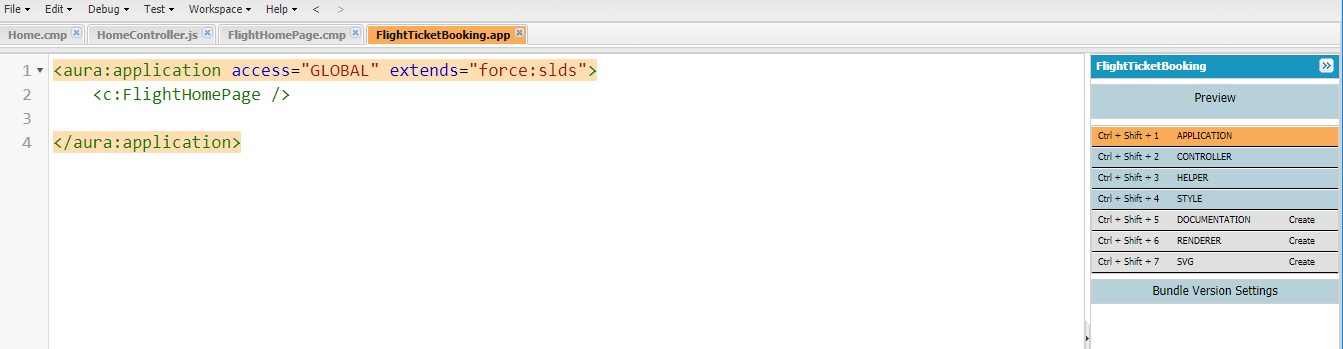
1. FlightHomePage.cmp

This is main component which holds the others three component. Makes transaction using event to switch between.



1. FlightTicketBooking.app

To run lightning App we need to add component in this App.



1. **Writing custom JavaScript Controller:**
2. **Creating Apex classes:**
3. **Adding SLDS design:**

# Integration & Testing

//To Do

# Implementation

//To Do

# Operations & Maintenance

// To Do

sfahg