

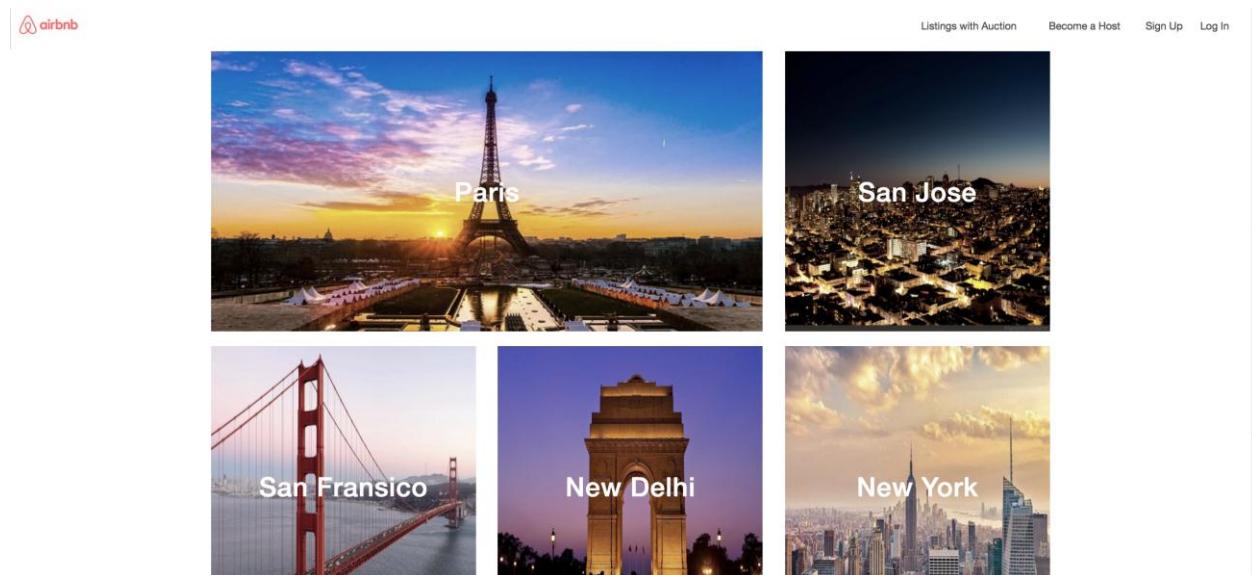
## Walkthrough:

### 1) HomePage:

This page provides user with options like SignUp and SignIn and Search Places.

The screenshot shows the top navigation bar with the Airbnb logo, a search bar containing 'Listings with Auction', and links for 'Become a Host', 'Sign Up', and 'Log In'. Below this is a large red banner with the text 'Slide Airbnb Places' and a subtext 'Get lost in a city with insiders, like Woody.' A button labeled 'See what's New' is visible. To the right of the banner is a search bar with fields for 'Where', 'When', and 'Guests', and a red 'Search' button. The search bar has placeholder text: 'Destination, city, address', 'Check In', 'Check Out', and 'Guests' (with a dropdown arrow). The background of the page features a collage of travel-related images, including a book titled 'NICKI & PAM ZEN EXPLORERS', a movie poster for 'VIRGO SOUND SYNTHE SUPREME', and a person running.

2) All the cities at home page. When customer clicks on any of the cities he is redirected to next page which shows the property under that city or user can search the city from above search bar.



3) Sign Up:

Kunal

Ahuja

kunal318@gmail.com

.....

I'd like to receive coupons, promotions, surveys, and updates via email about Airbnb and its partners.

By signing up, I agree to Airbnb's Terms of Service, Nondiscrimination Policy, Payments Terms of Service, Privacy Policy, Guest Refund Policy, and Host Guarantee Terms.

Sign Up

Already have an Airbnb account? [Log in](#)

	When	Guest
	Check In → Check Out	

Validations:

If password is less than 8 characters

The screenshot shows the Airbnb sign-up process. The user has entered "Kunal" for the first name, "Ahuja" for the last name, and "kunal318@gmail.com" for the email address. The password field contains ".....". A red error message at the bottom of the form states: "Your password must be at least 8 characters. Please try again." To the right of the error message is a small red 'X' icon. Below the password field is a checkbox for receiving newsletters: "I'd like to receive coupons, promotions, surveys, and updates via email about Airbnb and its partners." A note below the checkbox states: "By signing up, I agree to Airbnb's Terms of Service, Nondiscrimination Policy, Payments Terms of Service, Privacy Policy, Guest Refund Policy, and Host Guarantee Terms." At the bottom of the form is a large pink "Sign Up" button. Below the form, there is a link for existing users: "Already have an Airbnb account? Log in".

Kunal

Ahuja

kunal318@gmail.com

.....

Your password must be at least 8 characters. Please try again.

I'd like to receive coupons, promotions, surveys, and updates via email about Airbnb and its partners.

By signing up, I agree to Airbnb's Terms of Service, Nondiscrimination Policy, Payments Terms of Service, Privacy Policy, Guest Refund Policy, and Host Guarantee Terms.

Sign Up

---

Already have an Airbnb account? [Log in](#)

If EmailId is not formatted correctly(missing '@' from emailed)

Kunal

Ahuja

kunal318gmail.com

Please enter a valid email include @

.....

I'd like to receive coupons, promotions, surveys, and updates via email about Airbnb and its partners.

By signing up, I agree to Airbnb's Terms of Service, Nondiscrimination Policy, Payments Terms of Service, Privacy Policy, Guest Refund Policy, and Host Guarantee Terms.

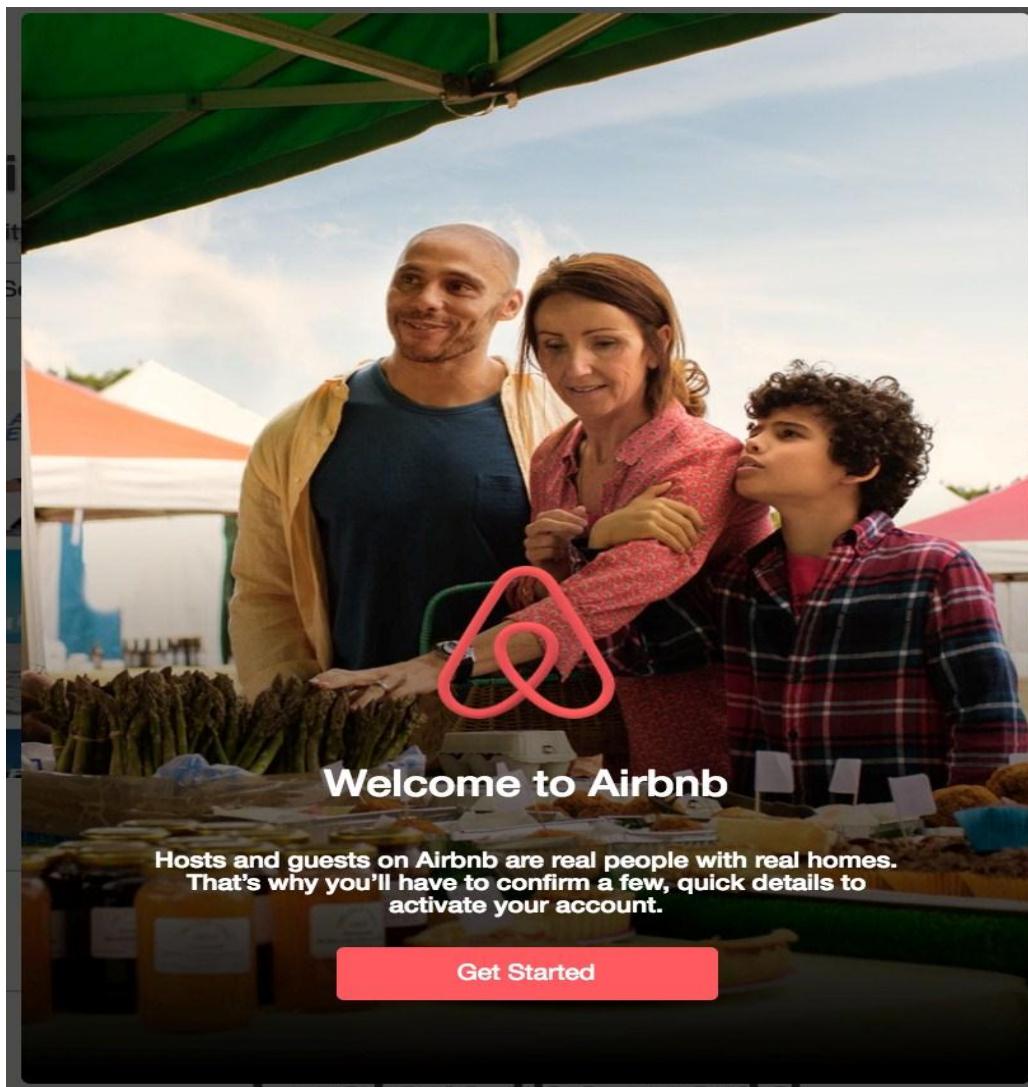
Sign Up

---

Already have an Airbnb account?

Log in

After successfully sign up user is asked to upload profile Pic:



## Add your profile photo

Put a face to your name! We'll add this to your profile, and share it with future hosts and guests.



Upload photo

## Add your profile photo

Put a face to your name! We'll add this to your profile, and share it with future hosts and guests.

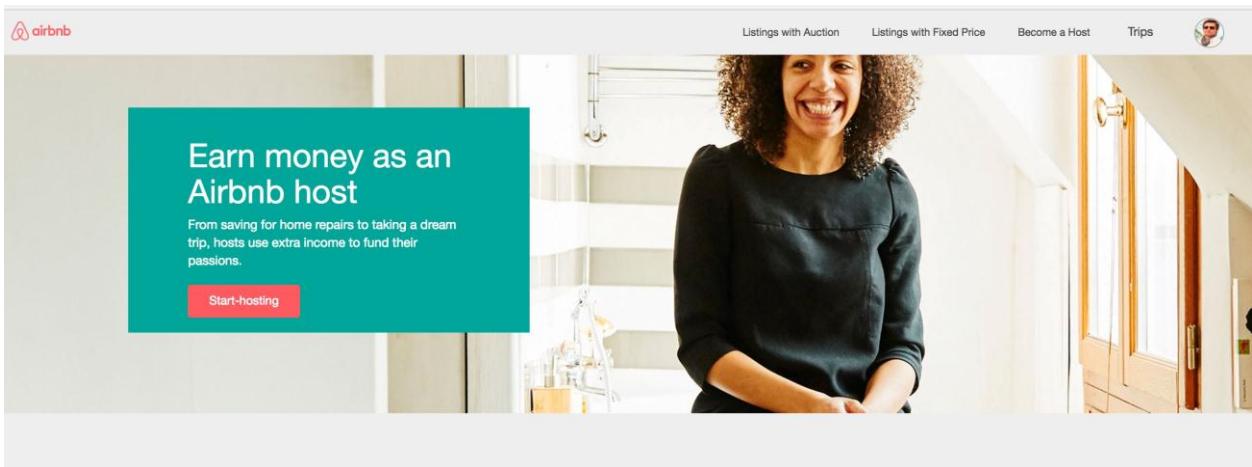


Change Photo

Upload Now

## Host Module:

When user wants to host his property:



User Enters the type of place:

**Hi, Kunal! Let's get you ready to become a host.**

STEP 1

**What kind of place do you have?**

Private room for 5 guest

San Jose

**How many bathrooms ?**

2 bathrooms - +

Conitnue

### Address Validation:

Since this Application's main business is renting places we have validated addresses using API known as **Address Validator**. If the address is incorrect we suggest the host right addresses as shown in below screenshot. User entered incorrect address we validated from Map and gave him the correct suggestions.

**Did you mean!**

329 North 1st Street, San Jose, CA, US

**Where's your place located?**

Country

United States

Street Address

329 North 1st Street

Apt, Suite, Bldg. (optional)

2144

City

San Jose

State

California

ZIP Code

95110

**Next**

**Code:**

```
module.exports.addressValidator =function(req, res) {  
var json_responses = '';  
address = req.body.address + ", " + req.body.city + ", " + req.body.state + ", " +  
req.body.country;  
console.log("Address " + address);  
var unknownAddresses = [];  
addressValidator.validate(address, addressValidator.match.unknown, function (err,  
exact, inexact) {  
    if (err) {  
        console.log('in error');  
        json_responses = {'statusCode': '401', 'error': 'Please enter valid data'};  
    } else {  
        if (exact != '' && exact != null) {  
            console.log('in match found');  
            json_responses = {'statusCode': '200', 'addressCoordinate': exact[0].location};  
        } else {  
  
            unknownAddresses = _.map(inexact, function (a) {  
return a.toString();  
});  
            console.log('in no match found');  
json_responses = {'statusCode': '400', 'suggestedAddress':  
unknownAddresses};  
    }  
});  
res.json(json_responses);  
};
```

User can select amenities available in his property:

[Listings with Auction](#)[Listings with Fixed Price](#)[Become a Host](#)[Trips](#)

### What amenities do you offer?

Essentials Towels, bed sheets, soap, and toilet paper

Wifi

Shampoo

Closet/drawers

TV

Heat

Air Conditioning

Iron

Safety Amenities

Smoke Detector

First Aid and Kit

[Next](#)

### Name of property:

[Listings with Auction](#)[Listings with Fixed Price](#)[Become a Host](#)[Trips](#)

Name your place

Gordon House

Enter a description

A beautiful Place with Swimming Pool

User can select if property is available for fixedPrice or Auction:

[Listings with Auction](#)[Listings with Fixed Price](#)[Become a Host](#)[Trips](#)

Auction

Fixed Price

Per day rent of the property and available dates

Airbnb

Listings with Auction Listings with Fixed Price Become a Host Trips

Auction Fixed Price

Fixed Price for your place

433

Set your Calendar

Available From Date Available Till Date

12/04/2016 12/21/2020

If user selects available from date less than current date message is thrown:

Airbnb

Listings with Auction Listings with Fixed Price Become a Host Trips

Auction Fixed Price

Fixed Price for your place

433

Set your Calendar

Available From Date Available Till Date

11/10/2016 12/21/2020

localhost:3000 says:  
Available from date should be greater than today's date

OK

User can upload images of his property. We have restricted upload image limit to 24 following Airbnb.

Show travelers what your space looks like



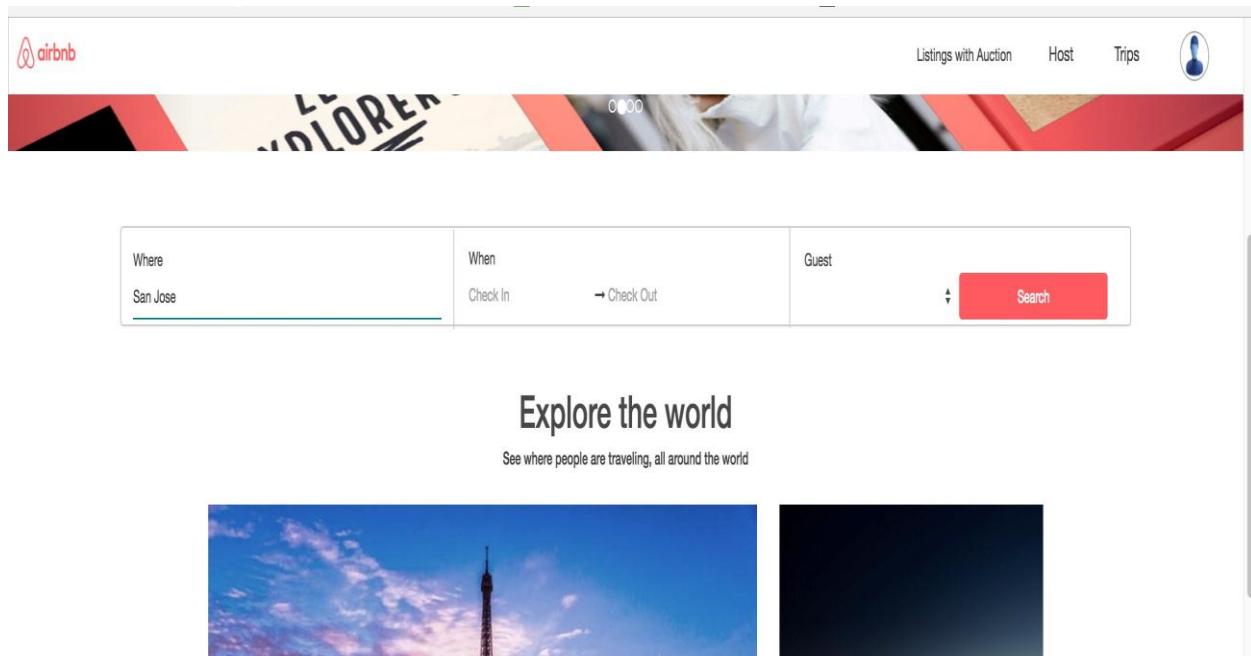
[Remove and select other from computer](#)

[Upload Now](#)

Now after this request of this property is sent to Admin for approval. Once admin approves then only property will be visible to other users for booking.

## Trip Module:

User can click on any of the cities or search for the properties in the city:



The screenshot shows the Airbnb search interface. At the top, there's a navigation bar with the Airbnb logo, a search bar containing "San Jose", and buttons for "Listings with Auction", "Host", "Trips", and a user profile icon. Below the search bar is a banner with the word "EXPLORE" and a price of "0.00". The main search form has three fields: "Where" (San Jose), "When" (Check In and Check Out), and "Guest" (dropdown menu). A red "Search" button is on the right. Below the search form is a section titled "Explore the world" with the sub-instruction "See where people are traveling, all around the world". It features two images: one of the Eiffel Tower at sunset and another dark, blurry image.

User can apply for the filters from the below screen. User sees all the properties. All the filters are function and are applied on click.

The screenshot shows the Airbnb search interface. At the top, there are date input fields ('mm/dd/yyyy') and a dropdown for 'Room type' with options 'Entire Home', 'Private Room', and 'Shared Room'. Below these are price range sliders for 'Price range' from \$20 to \$2000. A 'Filter' button is located below the sliders. On the right, there are links for 'Listings with Auction', 'Listings with Fixed Price', 'Host', and 'Trips'. A user profile icon is at the top right.

Below the filters, two property cards are displayed:

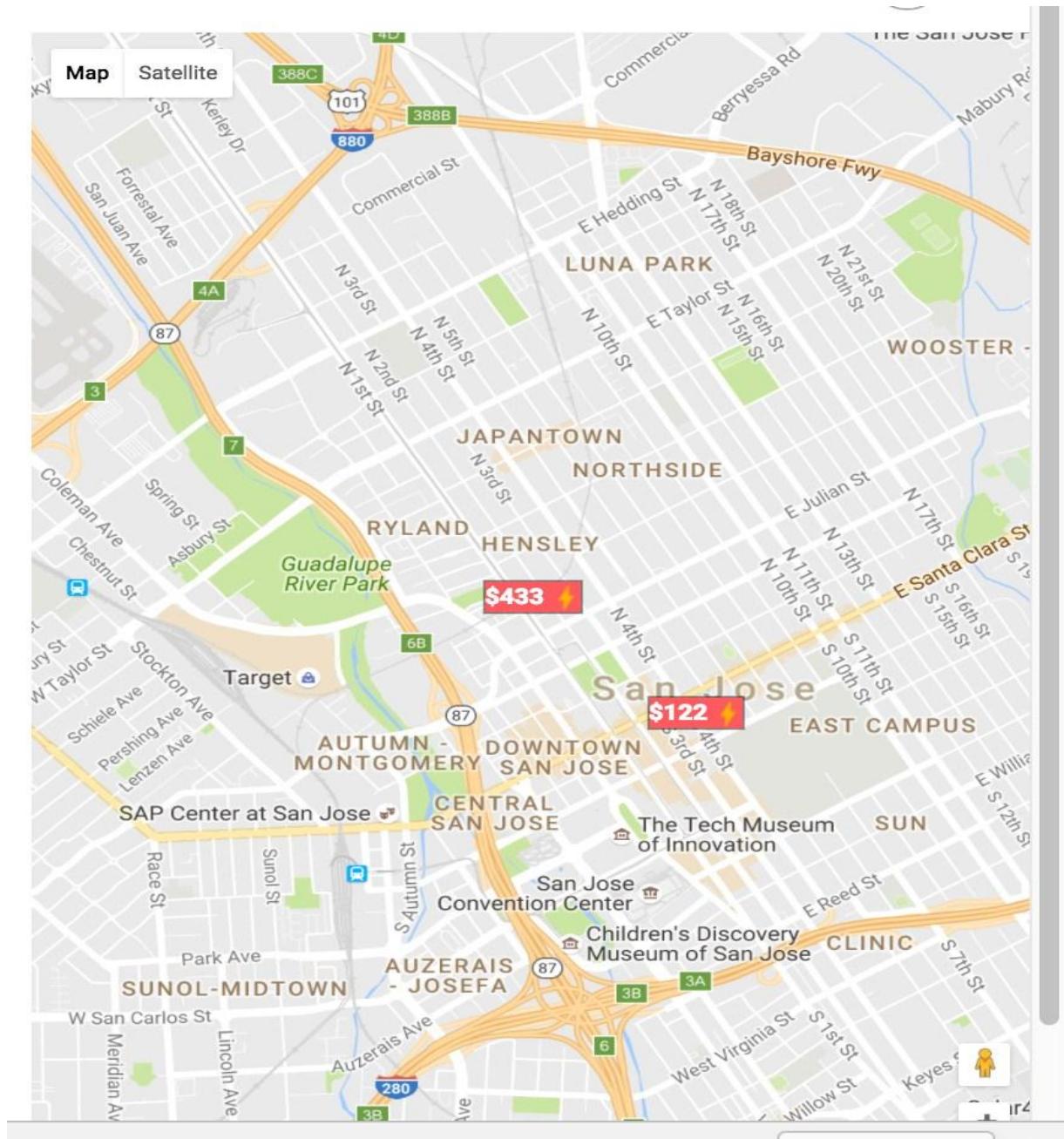
- Gordon House**: Private room • 5 guests • ★★★★☆. It shows a night view of a colorful town by a lake with a minaret, and a price of \$433.
- Sunshine Villas**: Shared room • 4 guests • ★★★★☆. It shows a modern house with a pool and a sign that says 'BEST HOUSES AUSTRALIA SEASON 1', with a price of \$122.

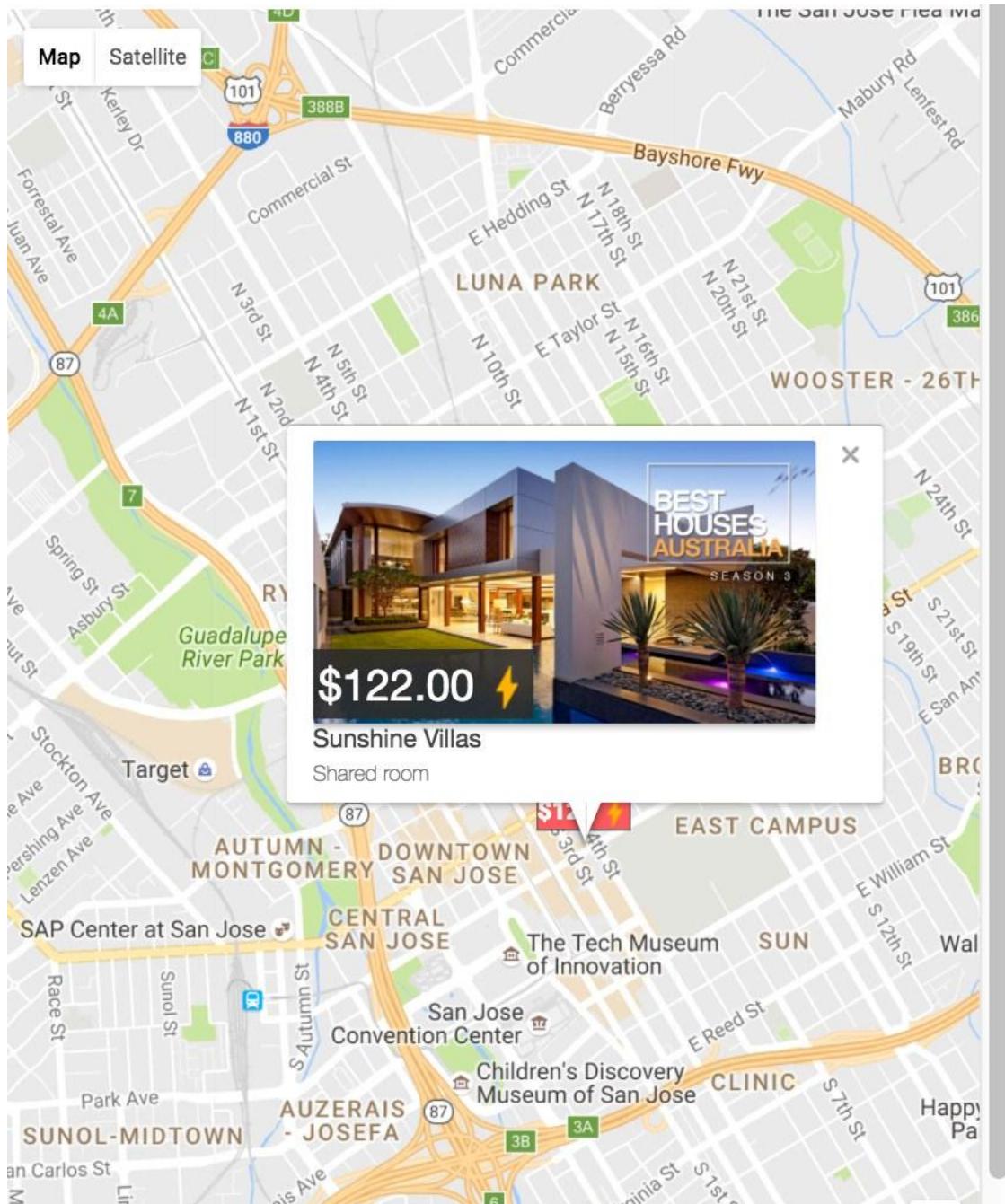
At the bottom right, a map of San Jose, California, is shown with several listing pins. One pin for 'Gordon House' is highlighted with a callout showing the price of \$433. Other pins are labeled with prices like \$122 and \$172. The map includes labels for Luna Park, San Jose Convention Center, and various neighborhoods.

user can also see the ratings for each and every property.

### Google Map:

User can see prices of each property on Map and those objects are clickable providing images of that property

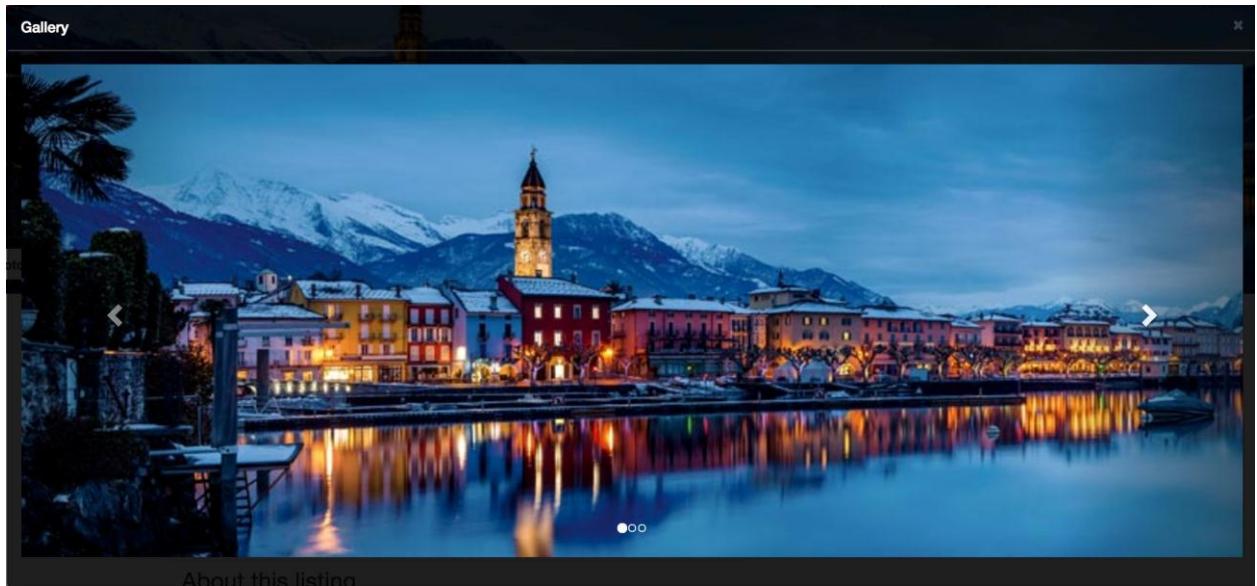




Once user clicks on any property he can view all the details of the property. He can select the checkIn and checkout dates.

The screenshot shows an Airbnb listing for "Gordon House" in San Jose, California. The main image is a night view of a town by a lake with snow-capped mountains in the background. The listing price is \$433 per night. It includes a private room for 5 guests with 2 bedrooms and 2 beds. The listing is 100% refundable. There is a "Book" button and a note that you would not be charged yet.

User can view all images of the properties. We have used carousel to display images.



User can see detailed review of the properties by other users along with the real images of the properties posted by the users.

The screenshot shows an Airbnb listing page. At the top, there's a navigation bar with links for 'Listings with Auction', 'Listings with Fixed Price', 'Host', 'Trips', and a user profile icon. Below the navigation, there's a section for 'CHECK-IN CHECK-OUT' with specific times listed. Underneath this, there's a 'Availability' section stating '1 night minimum stay'. A 'Reviews' section follows, showing '3 Reviews' with a 5-star rating. Below the reviews, there are breakdowns for Accuracy, Communication, Cleanliness, Location, Check in, and Value, each with a 5-star rating. A review from a user named 'Denis' is displayed, with a small profile picture of him. The review text reads: 'A great apartment in a great location. The metro was very close and you were not far from Romes major attractions. The apartment was spacious, wifi worked perfectly and there were plenty of places to eat and shop nearby. Claudio was helpful and reliable.'

We have applied all possible validations for selecting dates:

**\$433 ⚡** Per Night

Check In      Check Out      Guests

12/06/2016    12/01/2016    2

**Book**

CheckIn date should be before CheckOut date

You would not be charged yet

**\$433 ⚡** Per Night

Check In      Check Out      Guests

12/01/2016    12/01/2016    2

**Book**

CheckIn date and CheckOut date cannot be same

You would not be charged yet

**\$433 ⚡** Per Night

Check In      Check Out      Guests

11/03/2016    12/01/2016    2

**Book**

Please select a date after current date

You would not be charged yet

Since user has selected whilelisting his property that property is available only after 2016-12-06 and same goes for checkout validation as well.

**\$433** ⚡ **Per Night**

Check In  Check Out  Guests

**Book**

CheckIn date should be after 2016-12-06

You would not be charged yet

Once user selects the correct dates, he is redirected to final page where he can leave comments for the host and once he sends the request, request goes to approval for host and user is redirected to 'view trips' page.

**ABOUT YOUR TRIP**

Gordon House A beautiful Place with Swimming Pool, San Jose



Your host, Kunal

Who's coming?

2 Maximum Guests Allowed: 5

Say hello to your host and tell them why you're coming:

Hey, Please reserve this for me.

**Kunal ,House Rules**

- No Smoking ✓
- No parties ✓

Hosted by Kunal	2 Bedroom Apartment A beautiful Place with Swimming Pool
Private room	329 North 1st Street, San Jose, California, 95110, United States
Check-in 2016-12-20	> Checkout 2016-12-28
Accommodation	\$433.00
<b>Total</b>	<b>\$3,464.00</b>

## Booking Successful.

**About Your Trip**

Gordon H

**localhost:3000 says:**  
Booking Successfull

Prevent this page from creating additional dialogs.

**OK**

Who's coming?

2 Maximum Guests Allowed: 5

Say hello to your host and tell them why you're coming:

Hey, Please reserve this for me.

**Kunal ,House Rules**

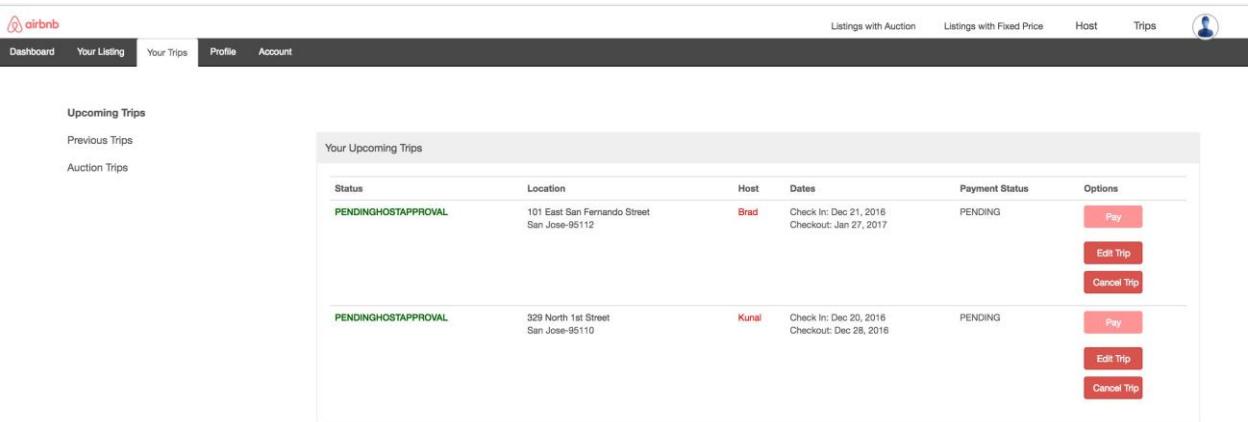
- No Smoking ✓
- No parties ✓
- Check-in Time : 01:00PM ✓
- Checkout Time : 12:00PM ✓

**Send Request**

Listings with Auction	Listings with Fixed Price
Host	Trips

## View Trips

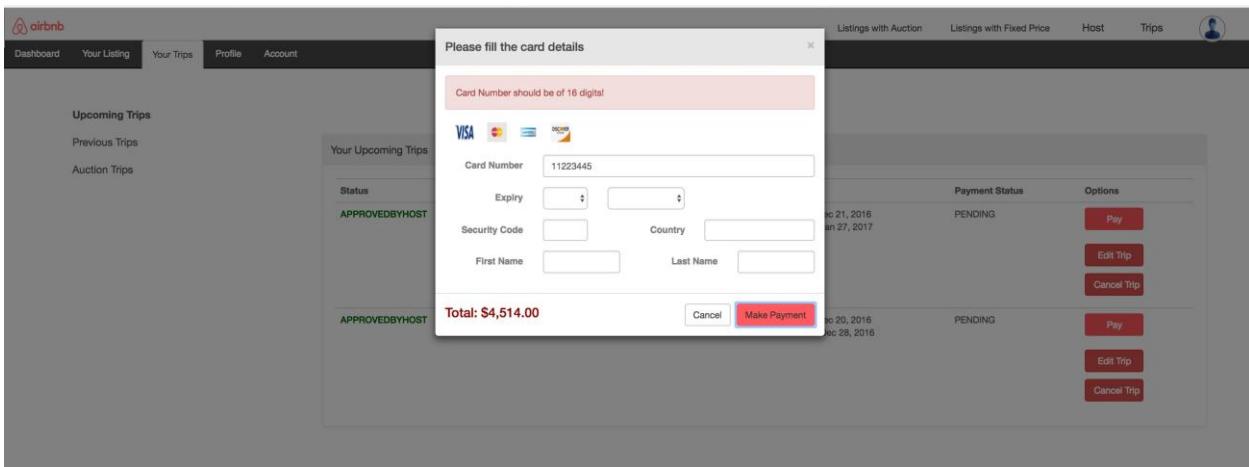
Unless hosts approves it user does not gets to pay and user can edit and cancel trip. Similarly user can see his previous trips and trips which he bid for.



The screenshot shows the Airbnb 'Your Trips' page. At the top, there are navigation links: Dashboard, Your Listing, Profile (which is selected), Account, Listings with Auction, Listings with Fixed Price, Host, Trips, and a user profile icon. On the left, there are three tabs: Upcoming Trips (selected), Previous Trips, and Auction Trips. The main area is titled 'Your Upcoming Trips' and displays two rows of trip information. Each row includes a status column ('PENDINGHOSTAPPROVAL'), a location column ('101 East San Fernando Street, San Jose-95112' and '329 North 1st Street, San Jose-95110'), a host column ('Brad' and 'Kunal'), a dates column ('Check In: Dec 21, 2016 / Checkout: Jan 27, 2017' and 'Check In: Dec 20, 2016 / Checkout: Dec 28, 2016'), a payment status column ('PENDING'), and three buttons: 'Pay', 'Edit Trip', and 'Cancel Trip'.

If admin approves the request, Pay button will be enabled and user can pay the amount. Once user clicks on Pay he gets an option to pay the amount. If user has already added the credit card in his profile the fields are already populated:

We have also applied the checks, credit card number should be 16 digits, cvv should be 3 digits and expiry should be greater than today's date.



User can view the bill once he pays the amount. View Bill Button is made visible after paying.

A screenshot of the Airbnb website showing the "Upcoming Trips" section. The header includes the Airbnb logo and navigation links: Dashboard, Your Listing, Your Trips (selected), Profile, Account, Listings with Auction, Listings with Fixed Price, Host, Trips, and a user profile icon. The main content area shows a table titled "Your Upcoming Trips". The table has columns: Status, Location, Host, Dates, Payment Status, and Options. There are two rows of data:

- The first row is for a trip "APPROVEDBYHOST" at "101 East San Fernando Street, San Jose-95112" hosted by "Brad". The dates are "Check In: Dec 21, 2016" and "Check Out: Jan 27, 2017". The payment status is "PAID". The "Options" column contains a "Pay" button (highlighted in red) and a "View Bill" button (highlighted in blue).
- The second row is for a trip "APPROVEDBYHOST" at "329 North 1st Street, San Jose-95110" hosted by "Kunal". The dates are "Check In: Dec 20, 2016" and "Check Out: Dec 28, 2016". The payment status is "PENDING". The "Options" column contains an "Edit Trip" button and a "Cancel Trip" button.

## Billing

Our Bill has covered all the fields like property name, Receipt number(ssn format), dates, amount, host, guests etc..

## Customer Receipt

Confirmation Code: 590

Wed March 7, 2012

Receipt # 590-31-7835

Airbnb  
+1 (669) 263 7801  
[WWW.AIRBNB.COM/CONTACT](http://WWW.AIRBNB.COM/CONTACT)

 Kunal

NAME

 101 East San Fernando Street

ACCOMODATION ADDRESS

 brad@gmail.com

HOST

 2016-12-21 01:00PM

CHECKIN

 37

NIGHTS

 San Jose

TRAVEL DESTINATION

 Kingdom Palace

PROPERTY

 2

GUESTS

 2017-01-27 12:00PM

CHECKOUT

 \$0

SECURITY DEPOSIT

## Payment Details

Accomodations	\$4,514.00(\$122.00 per night)
Service Fee	\$0
Total	\$4,514.00

Payment Received: Sun, May 01, 2016  
Card : XXXXXXXXXXXXXXXXX7788

\$4,514.00



Thanks for travelling with us

## User Profile:

A screenshot of the Airbnb profile edit page. The form fields include:

- First Name: Kunal
- Last Name: Smith
- I Am: Male
- Birth Date: 12/13/2016
- Email Address: kuna312@gmail.com
- Phone Number: 6692637801
- Preferred Language: English
- Preferred Currency: USD
- Street Address: 329 N 1st Street
- City: San Jose
- State: California
- Country: United States
- Zip Code: 95110
- Describe Yourself: I am a good host!

User can view his profile.

A screenshot of the Airbnb user profile page for "Kunal". The profile includes:

- Profile Picture: Placeholder for Kunal
- Name: Kunal
- View Profile
- Edit Profile
- Photos and Video
- Reviews
- User Details (Section):
  - First Name: Kunal
  - Last Name: Smith
  - Gender: male
  - Birth Date: 2016-12-13
  - Email Address: kuna312@gmail.com
  - Phone Number: 6692637801
  - Preferred Language: English
  - Preferred Currency: USD
  - Street Address: 329 N 1st Street
  - City: San Jose
  - State: California
  - Country: United States
  - Zip Code: 95110

User can upload his profile pic and video.

airbnb

Dashboard Your Listing Your Trips Profile Account

Listings with Auction Listings with Fixed Price Host Trips

View Profile

Edit Profile

Photos and Video

Reviews

**Profile Photo**

Images tell a lot about oneself, please upload a clear picture which will help people to learn about you!

*\*required*

**Profile Video**

You haven't recorded a video yet!

Video Checklist!

- We suggest that you sit at arm's length.
- You have got 30 seconds, so do your best to be concise.
- Try to keep the light in front of you.
- Look into the camera to create that personal connection.

*\*required*

airbnb

Dashboard Your Listing Your Trips Profile Account

Listings with Auction Listings with Fixed Price Host Trips

View Profile

Edit Profile

Photos and Video

Reviews

**Profile Photo**

Images tell a lot about oneself, please upload a clear picture which will help people to learn about you!

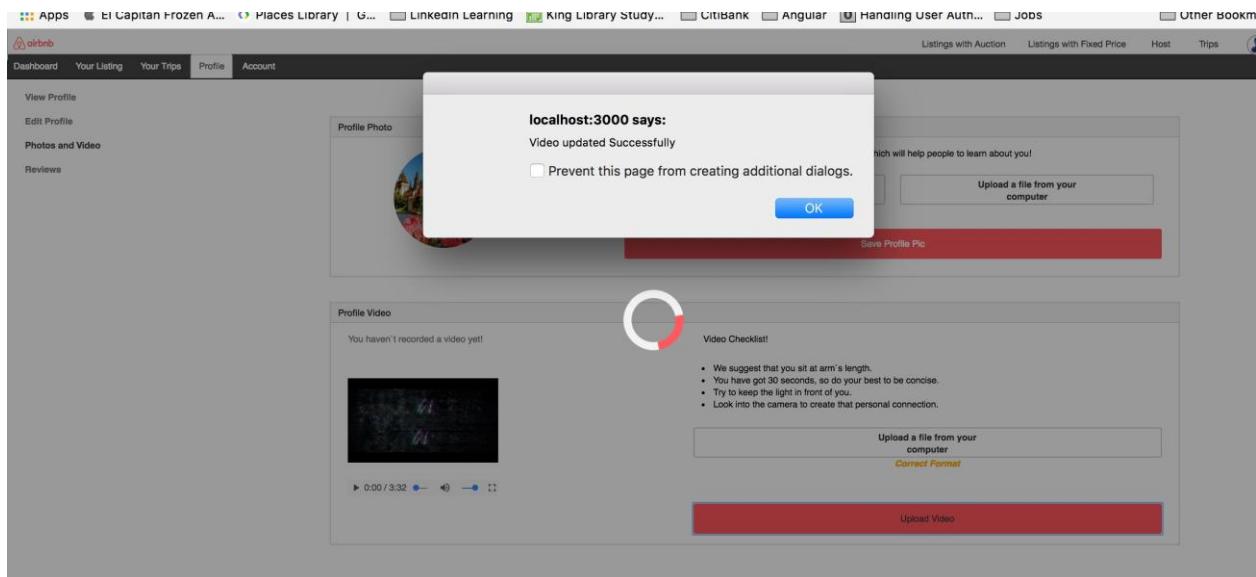
**Profile Video**

You haven't recorded a video yet!

Video Checklist!

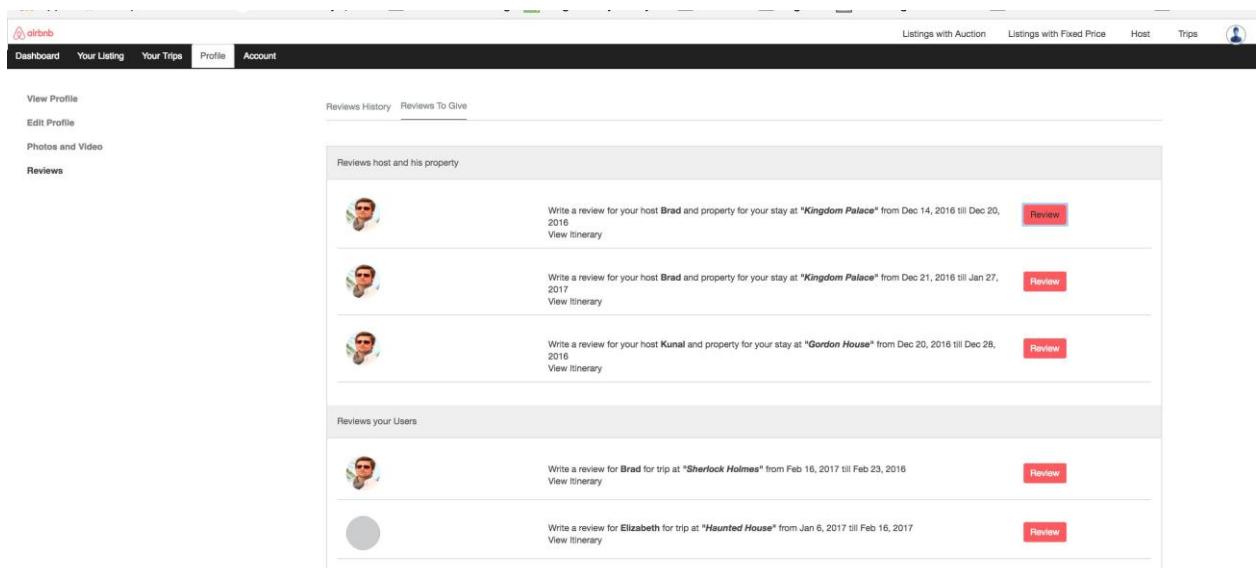
- We suggest that you sit at arm's length.
- You have got 30 seconds, so do your best to be concise.
- Try to keep the light in front of you.
- Look into the camera to create that personal connection.

*Correct Format*



## Reviews:

User can review the properties and host once the trip gets completed.





Review your host Brad for  
stay at property "**Kingdom  
Palace**"



x

**What you loved best about your  
stay**

Wonderful..

**Cleanliness Feedback**



Was very clean|

**Accuracy Feedback**



Accuracy of the place

**Location Feedback**



Location of the place

**Communication Feedback**



Communication of the host

**CheckIn Feedback**



CheckIn time of the place

**Value Feedback**



what you loved about the place

**Upload images for your review**

**Upload Images**

**Submit Review**

User can also upload real images:

*Was very clean*

**Accuracy Feedback**



*Accuracy of the place*

**Location Feedback**



*Location of the place*

**Communication Feedback**



*Communication of the host*

**CheckIn Feedback**



*CheckIn time of the place*

**Value Feedback**



*what you loved about the place*

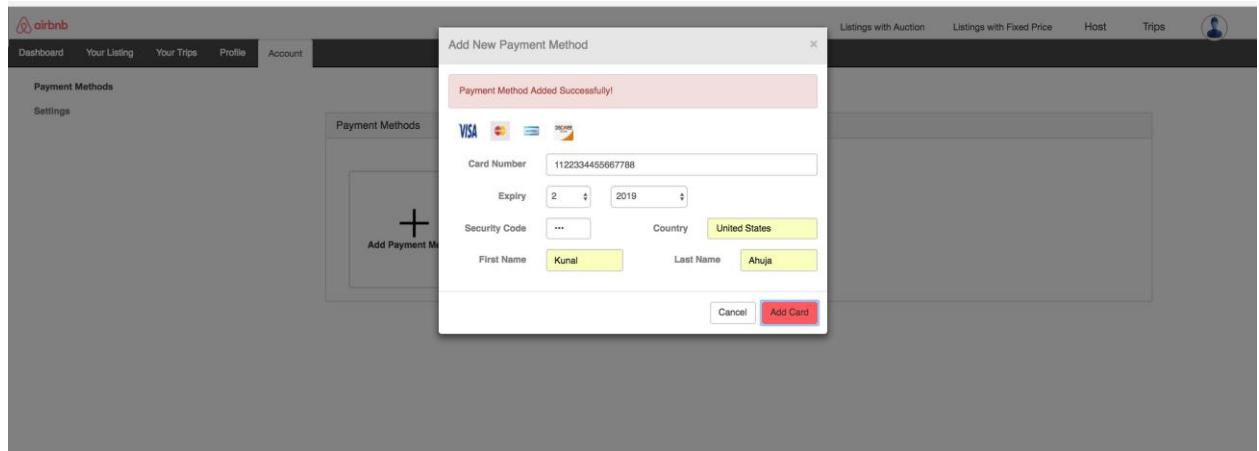
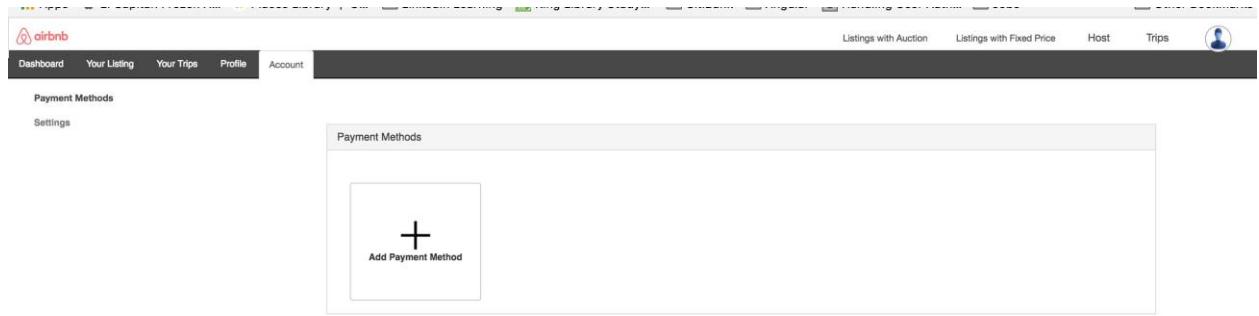
**Upload images for your review**



*Upload Images*

**Submit Review**

User can add his credit card details:



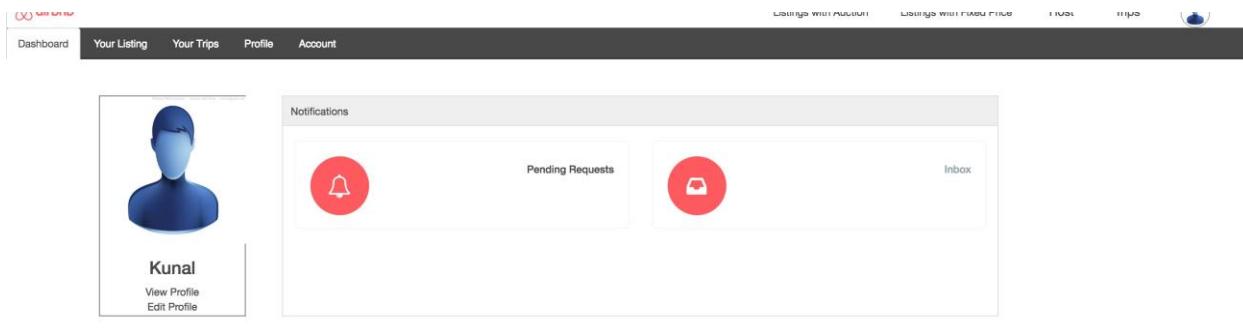
The screenshot shows the 'Payment Methods' section of the Airbnb account settings. At the top, there's a header with the Airbnb logo and navigation links: Dashboard, Your Listing, Your Trips, Profile, Account, Listings with Auction, Listings with Fixed Price, Host, Trips, and a user profile icon. Below the header, the 'Payment Methods' section has a 'Settings' link. The main area is titled 'Payment Methods' and contains two boxes: one for adding a new method (with a plus sign icon) and one showing an existing card (XXXXXX7788, Expiry Date: 2/2019, CardHolder: Kunal Ahuja).

User can manage his listings, check who have booked his properties.

The screenshot shows the 'Reservations' section of the Airbnb host dashboard. At the top, there's a header with the Airbnb logo and navigation links: Dashboard, Your Listing, Your Trips, Profile, Account, Listings with Auction, Listings with Fixed Price, Host, Trips, and a user profile icon. On the left, there are links for 'Your Listings', 'Your Reservations', and 'Manage Listing'. The main area is titled 'Reservations' and displays a table with two rows of trip details. The columns are: Status, Dates and Location, Guest, Details, and Payment-Details. The first row (Status: TRIPCOMPLETED) shows a guest named Brad (image), payment amount \$854, and card number XXXX-XXXX-XXXX-7788. The second row (Status: APPROVEDBYHOST) shows a guest named Elizabeth (image), payment amount \$14145, and card number XXXX-XXXX-XXXX-7788.

Status	Dates and Location	Guest	Details	Payment-Details
TRIPCOMPLETED	16.02.2017 -> 23.02.2016 Sherlock Holmes London, 43567	Brad	<button>Bill</button>	Amount: \$854 XXXX-XXXX-XXXX-7788
APPROVEDBYHOST	06.01.2017 -> 16.02.2017 Haunted House San Jose, 95110	Elizabeth	<button>Bill</button>	Amount: \$14145 XXXX-XXXX-XXXX-7788

Host Dashboard:



User can click on any pending requests to approve the requests:

Status	Dates and Location	Guest	Details	Actions
<span style="background-color: green; color: white; padding: 2px;">pendingHostApproval</span>	16.02.2017 -> 23.02.2016 Sherlock Holmes  London, 43567	Brad	\$854 total Message History Print Confirmation	<span style="border: 1px solid red; padding: 2px 5px;">Accept</span> <span style="border: 1px solid red; background-color: red; color: white; padding: 2px 5px;">Cancel</span>
<span style="background-color: green; color: white; padding: 2px;">pendingHostApproval</span>	06.01.2017 -> 16.02.2017 Haunted House  San Jose, 95110	John Elizabeth	\$14145 total Message History Print Confirmation	<span style="border: 1px solid red; padding: 2px 5px;">Accept</span> <span style="border: 1px solid red; background-color: red; color: white; padding: 2px 5px;">Cancel</span>

### Bidding:

User can place a bid selecting listing auctions:

**G house Bidding avail**

San Jose, CA, United States • ★★★★☆ 10 reviews

**\$112** Per Night

**Check In** 12/17/21 **Check Out** 02/03/21 **Guests** 2

**\$112 X 48 nights** \$5376

**Minimum Bid Amount** \$5376

**Bid Amount** \$ 6000

**Place Bid**

**Entire Place** **5 Guests** **2 Bedroom** **2 Beds**

**100% refundable**  
Cancel up to 30 days before your trip and get a full refund, including service fees.

## Admin Dashboard:



Pending Approval Listings for admin:

Dashboard

Listings

review

Pending Listings



Host: Kunal



on 12/4/16 12:00 AM  
List Id: 522-29-9133

House

agag - Shared room

Address:  
...

Listing Id: 522-29-9133  
Facility: Laundry – washer,  
Basic Amenities: Shampoo,  
Number of Bathrooms : 2  
From Date: Dec 7, 2016 12:00:00 AM - To Date: Sep 29, 2017 12:00:00 AM  
Price: 144 USD

Status: newListing

Accept Reject

## Approved Listing:

Dashboard

Listings

review

Approved Listing



Host: Kunal



Kunal  
on 12/3/16 12:00 AM  
List Id: 004-24-2784

Haunted House

Ghost house - Entire Place

Address:  
...

Listing Id: 004-24-2784  
Facility: Laundry – dryer,  
Basic Amenities: Air Conditioning, Iron,  
Number of Bathrooms : 2  
From Date: Dec 31, 2016 12:00:00 AM - To Date: Dec 31, 2018 12:00:00 AM

Admin can view the users:

The screenshot shows a user interface for managing users. On the left, there is a sidebar with navigation links: Dashboard, Listings, and review. The main area is titled "Review Users" and contains a section titled "User Details". It features a search bar with the placeholder "Enter City Name" and two buttons: "Search" (green) and "Reset grid" (red). Below the search bar is a table with the following columns: First Name, Last Name, UserId, Email, Contact, Host, City, Created on, and Profile. The table lists eight users with the following data:

First Name	Last Name	UserId	Email	Contact	Host	City	Created on	Profile
Kunal	Smith	503-24-0005	kunal312@gmail.com	6692637801	true	San Jose	12/3/16 12:00 AM	<button>View</button>
Elizabeth	Smith	047-46-4164	ebryantgr@cloudflare.com		true		12/3/16 12:00 AM	<button>View</button>
Frances	Smith	069-84-9108	fcole8u@yandex.ru		true		12/3/16 12:00 AM	<button>View</button>
Heather	Smith	503-56-4970	hlittle1n@nbcnews.com		true		12/3/16 12:00 AM	<button>View</button>
Richard	Smith	652-07-5959	rpowell@themeforest.net		true		12/3/16 12:00 AM	<button>View</button>
Bobby	Smith	646-56-8309	bhernandez96@usnews.com		true		12/3/16 12:00 AM	<button>View</button>
Jacqueline	Smith	496-52-2617	jdeanhf@php.net		true		12/3/16 12:00 AM	<button>View</button>
Craig	Smith	379-64-3372	chamiltonhg@bizjournals.com		true		12/3/16 12:00 AM	<button>View</button>

Admin can view the bills:

Dashboard

Review hosts

Listings

review

### Enter Bill date

mm/dd/yyyy

Search by date

### Search by Month and Year

Search by month

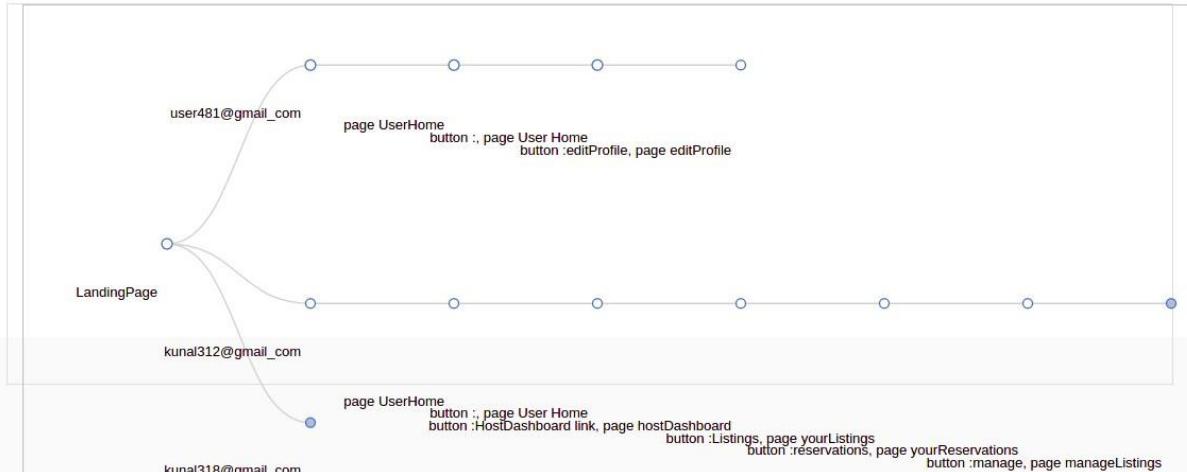
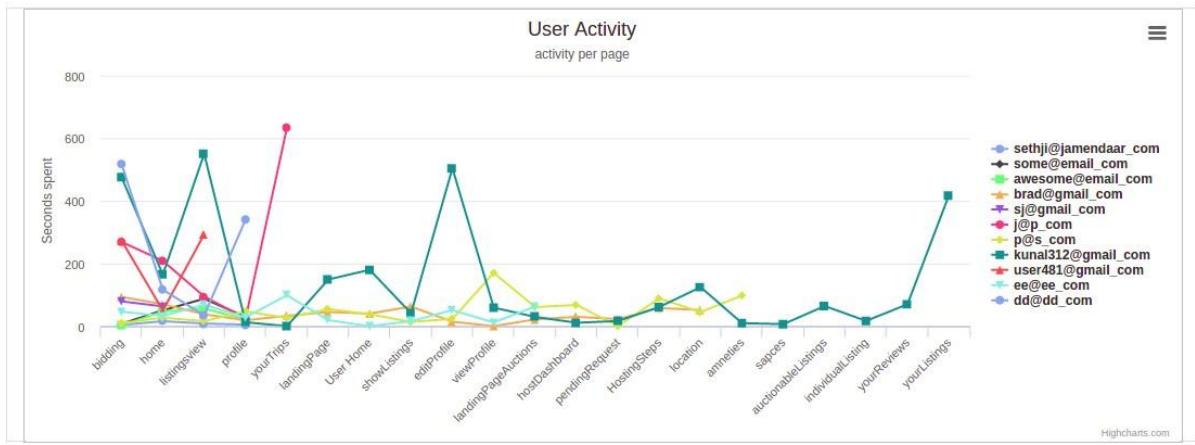
Reset grid

Bill Id	Bill Date	Host Name	Listing City	Payment Status	Trip Status	User Email	check In Date	check Out Date	View Bill
497-44-1823	12/2/16 12:00 AM	Kunal	London	paid	tripCompleted	brad@gmail.com	2/16/17 12:00 AM	2/23/16 12:00 AM	<a href="#">View</a>
445-25-2380		Brad	San Jose	cancelled	pendingHostApproval	kunal312@gmail.com	12/14/16 12:00 AM	12/20/16 12:00 AM	<a href="#">View</a>
590-31-7835	12/4/16 12:00 AM	Brad	San Jose	paid	pendinghostApproval	kunal312@gmail.com	12/21/16 12:00 AM	1/27/17 12:00 AM	<a href="#">View</a>
516-60-2431	12/4/16 12:00 AM	Kunal	San Jose	paid	pendingHostApproval	ebryantgr@cloudflare.com	1/6/17 12:00 AM	2/16/17 12:00 AM	<a href="#">View</a>

## Admin Analytics graphs:

### ❖ Admin Dashboard

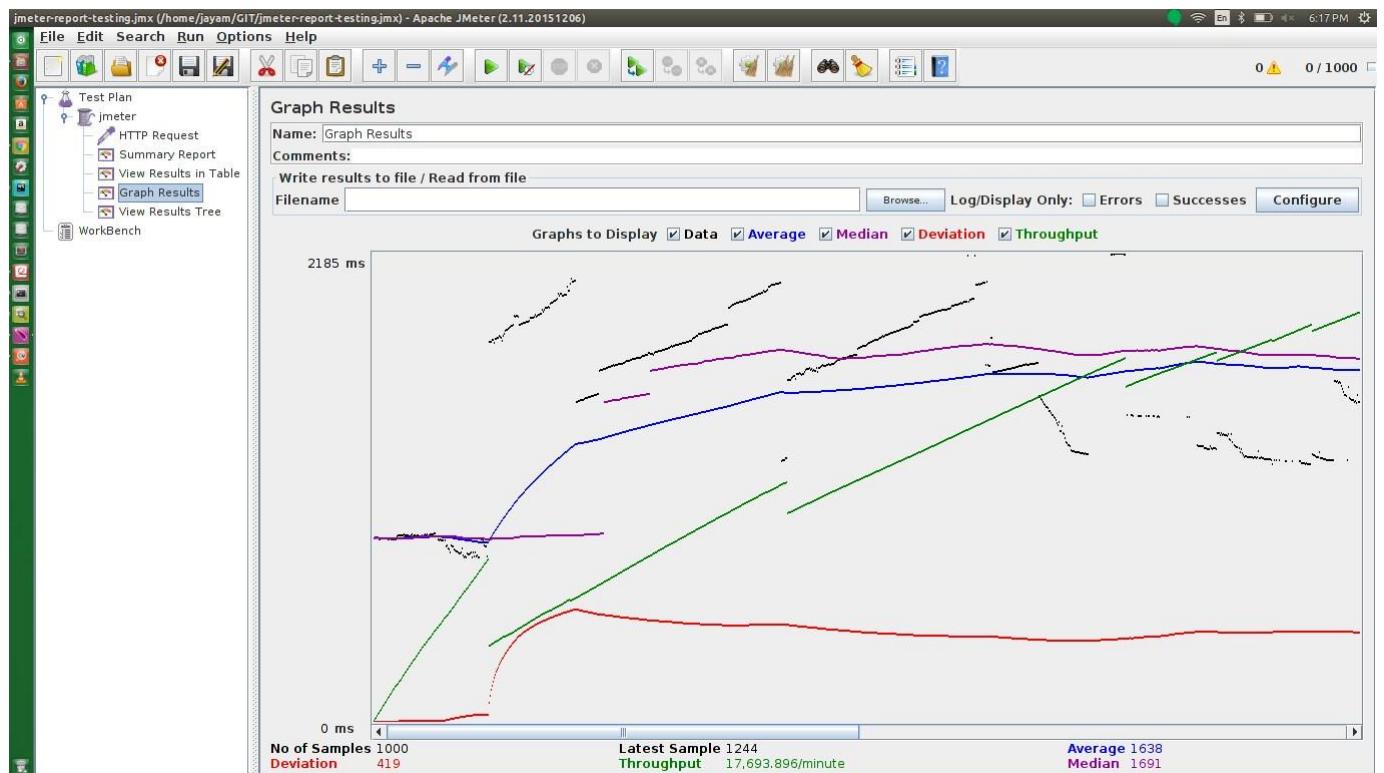




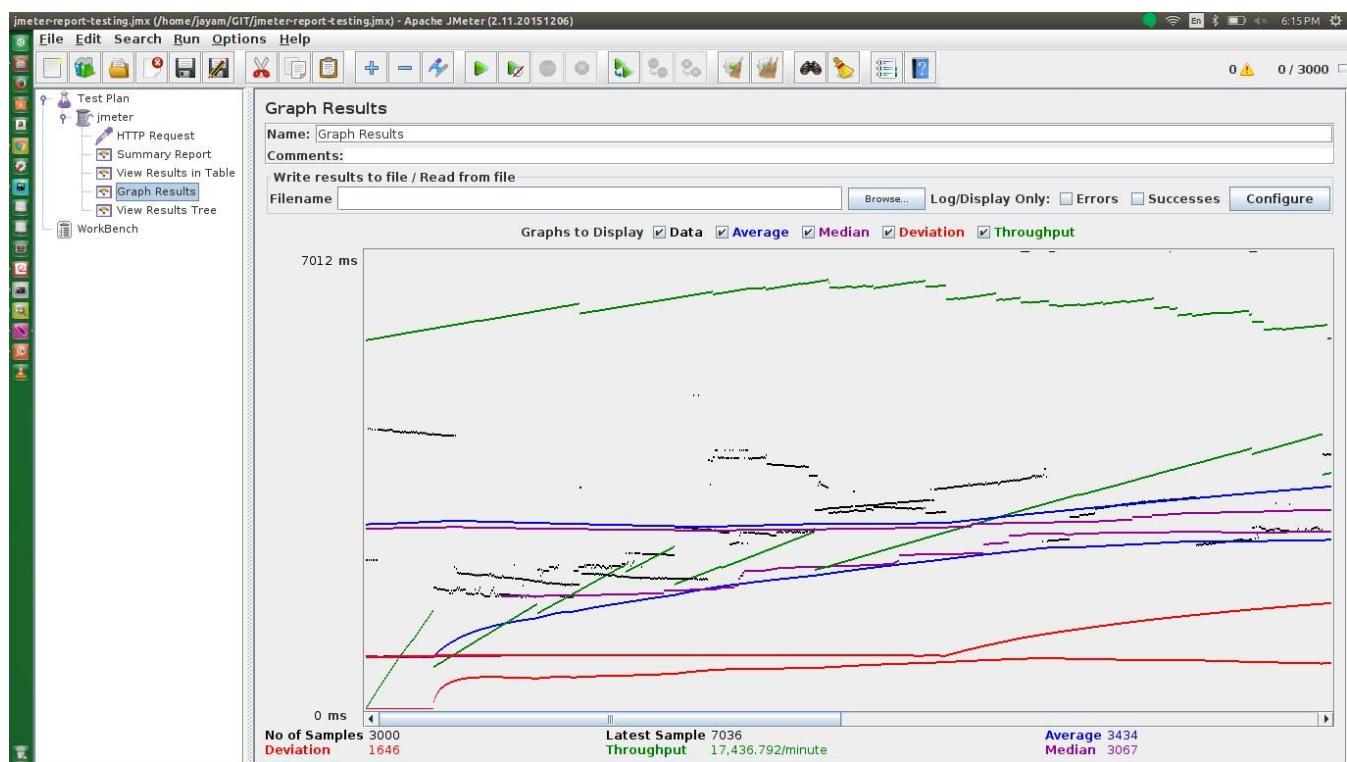
Testing:

❖ Jmeter Graphs

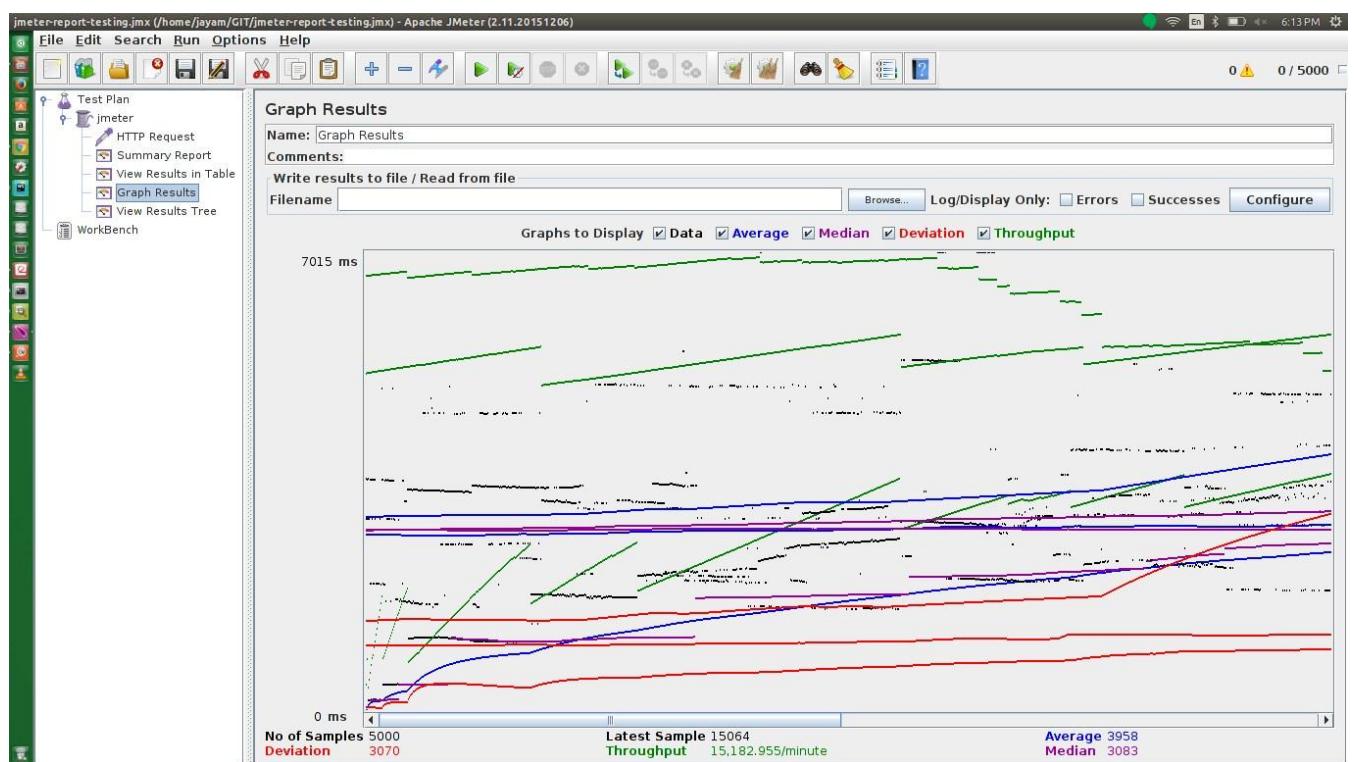
• 1000 Requests - Throughput	Base (Simple / minute Average)	Database	Connection
		-	17693.896



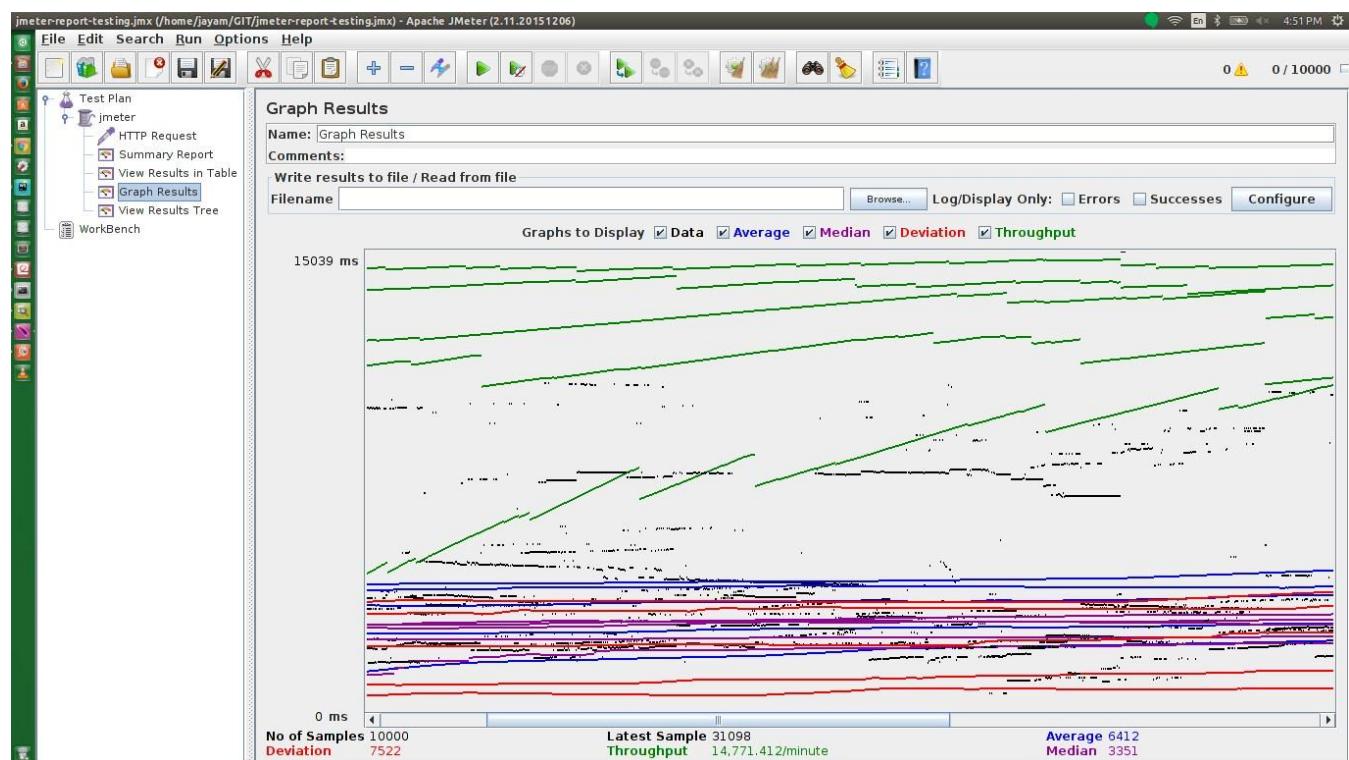
- **3000 Requests** - Base (Simple Database Connection)  
 Throughput / minute - 17436.792 Average -  
 3434



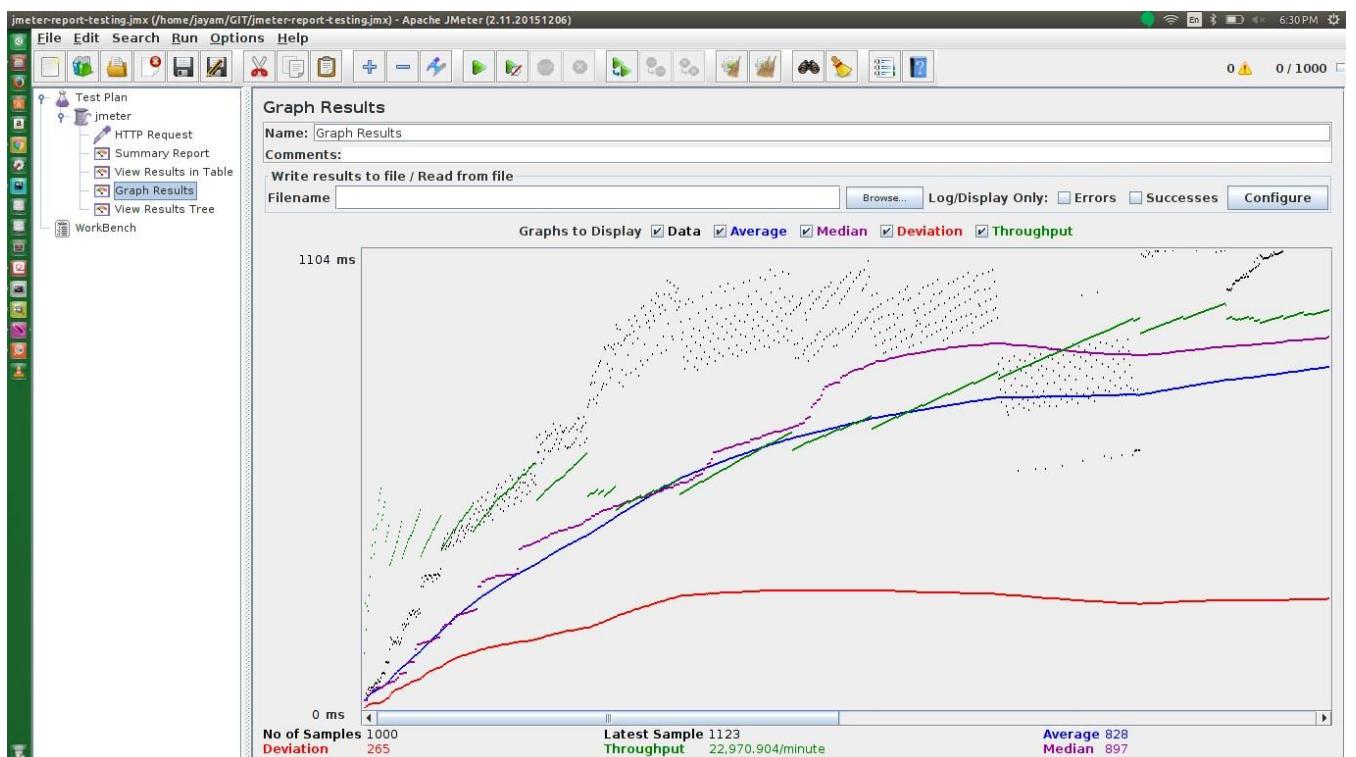
- **5000 Requests** - Base (Simple / minute) Database Connection) 15182.955
- Throughput -
- Average - 3958



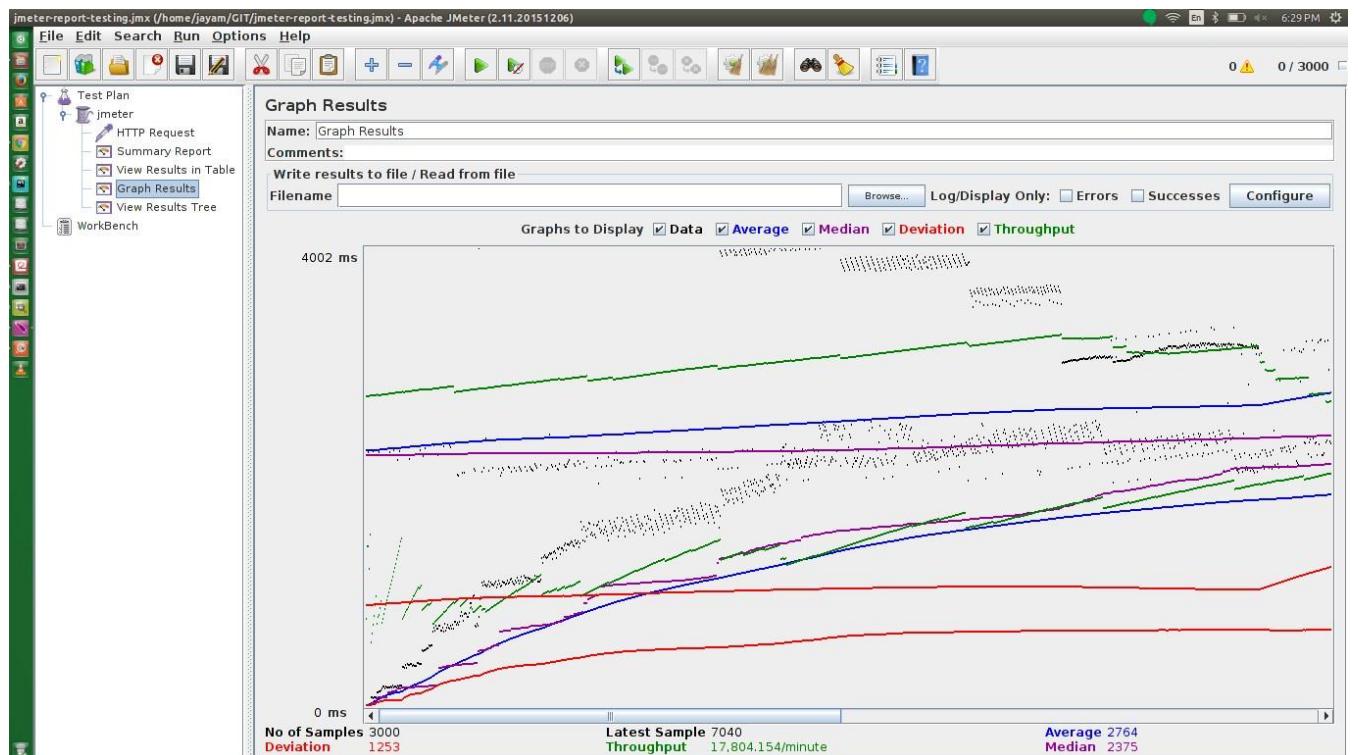
- **10000 Requests** - Base (Simple Database Connection)  
 Throughput / minute - 14771.412  
 - 6412



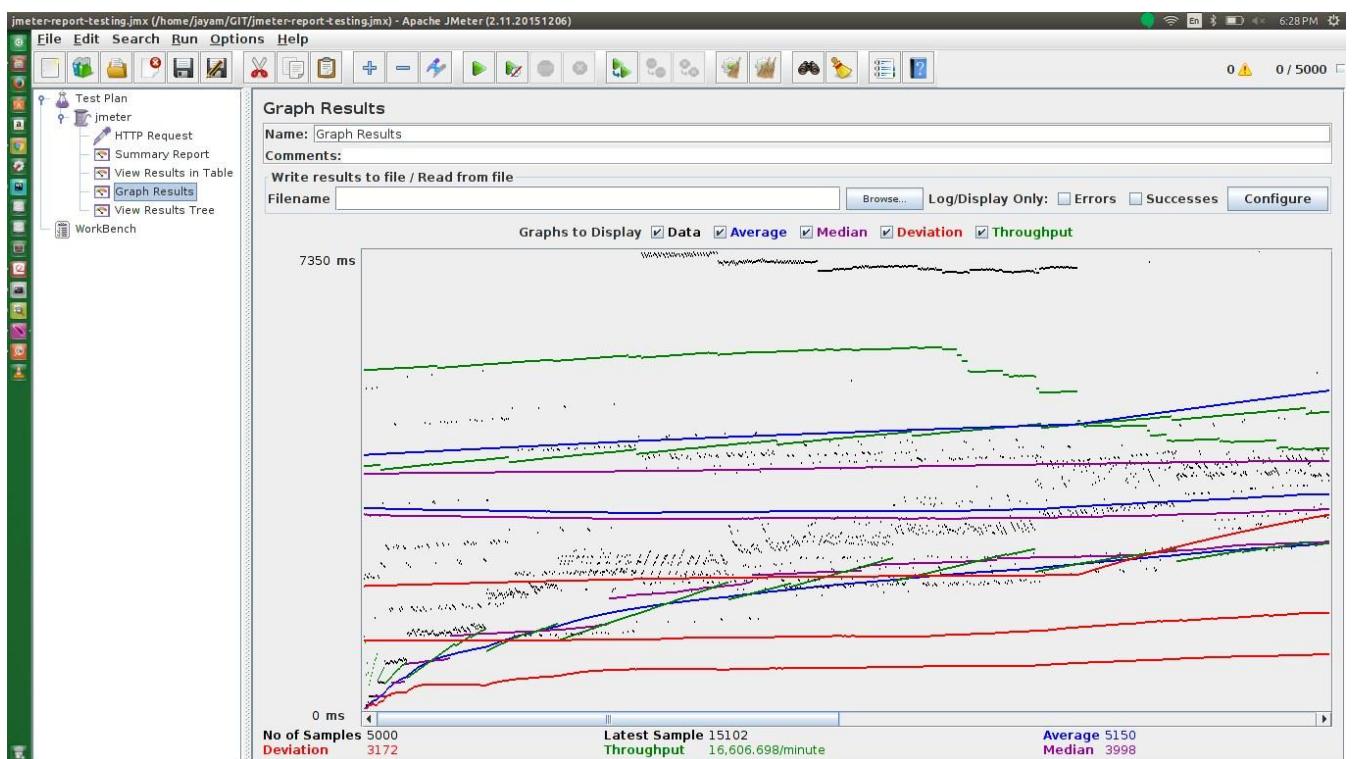
• **1000 Requests** - Base / Connection minute - Pooling  
 Throughput - / 828      22970.904      Average



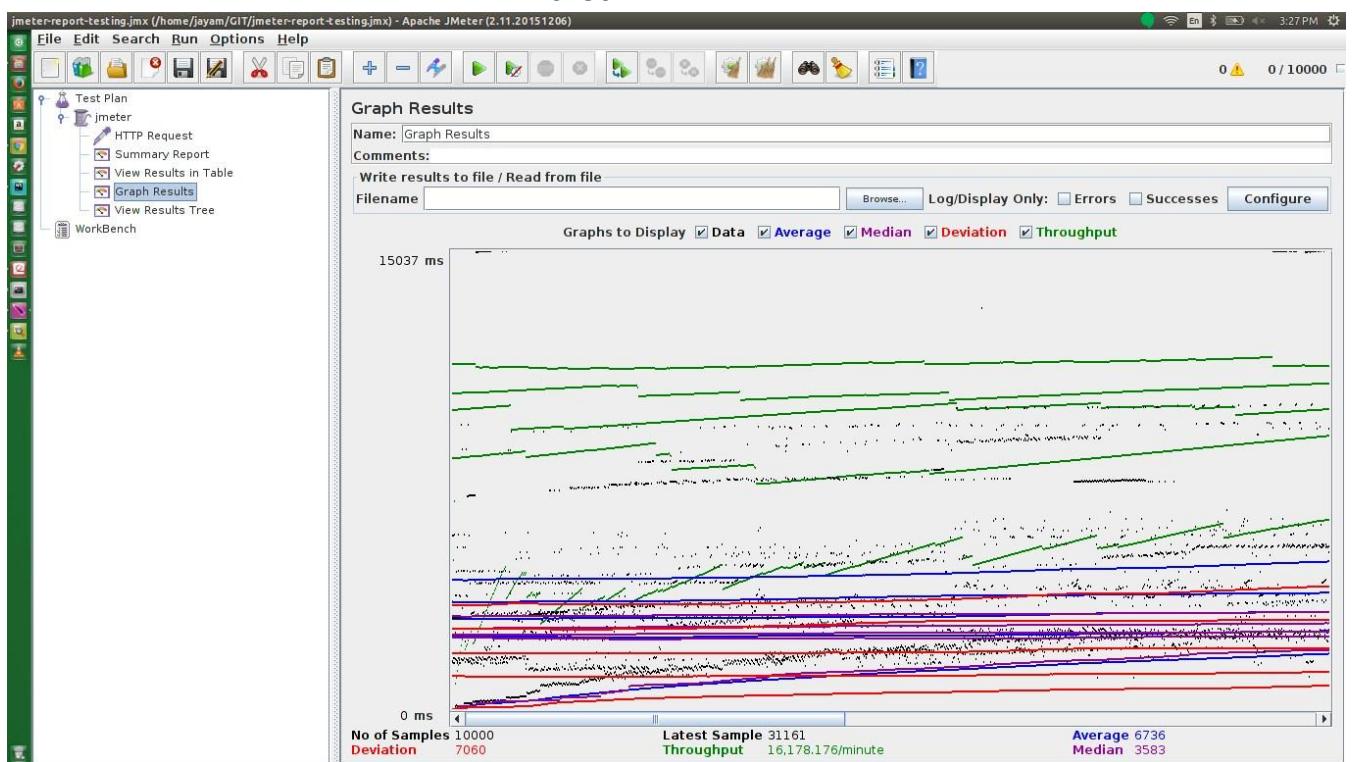
• **3000 Requests** - Base + Connection Pooling  
 Throughput / minute - 17804.154 Average  
 - 2764



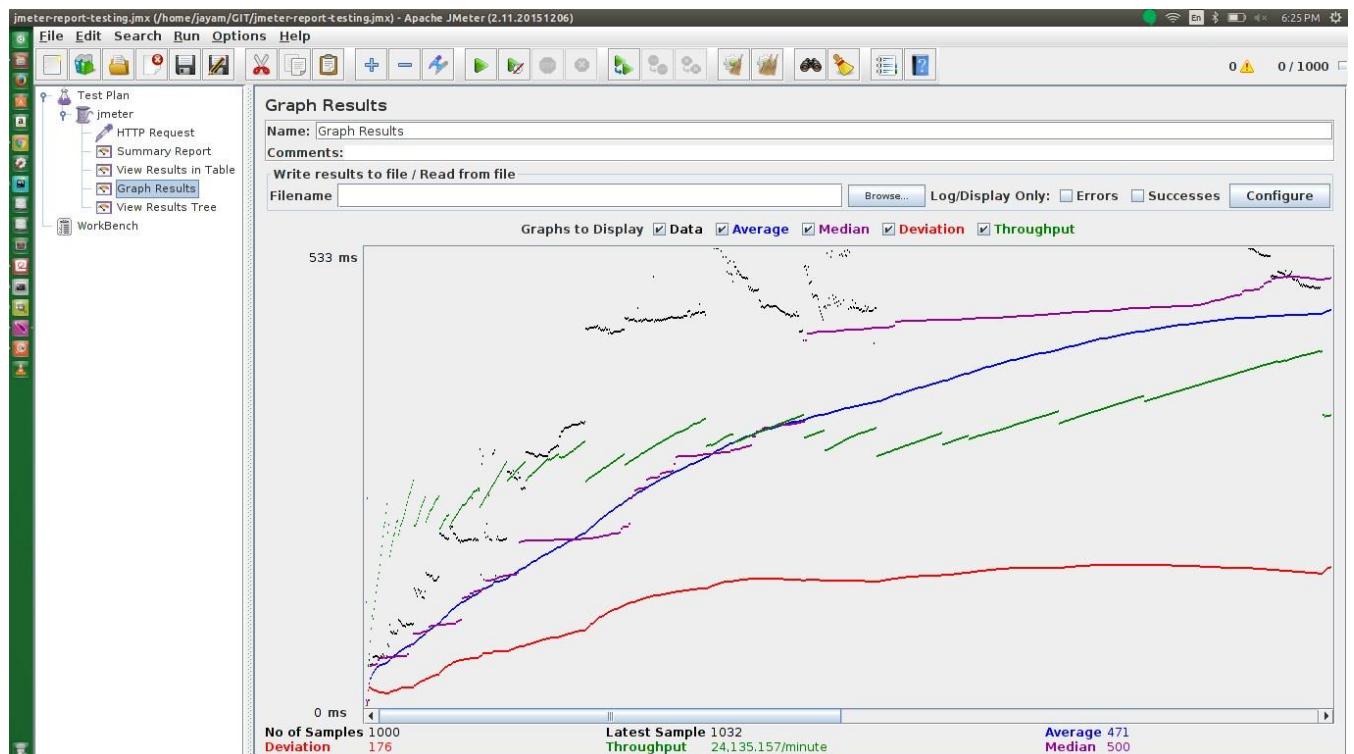
**• 5000 Requests** - Base / Connection Pooling  
 Throughput minute - 16606.698 Average  
 - 5150



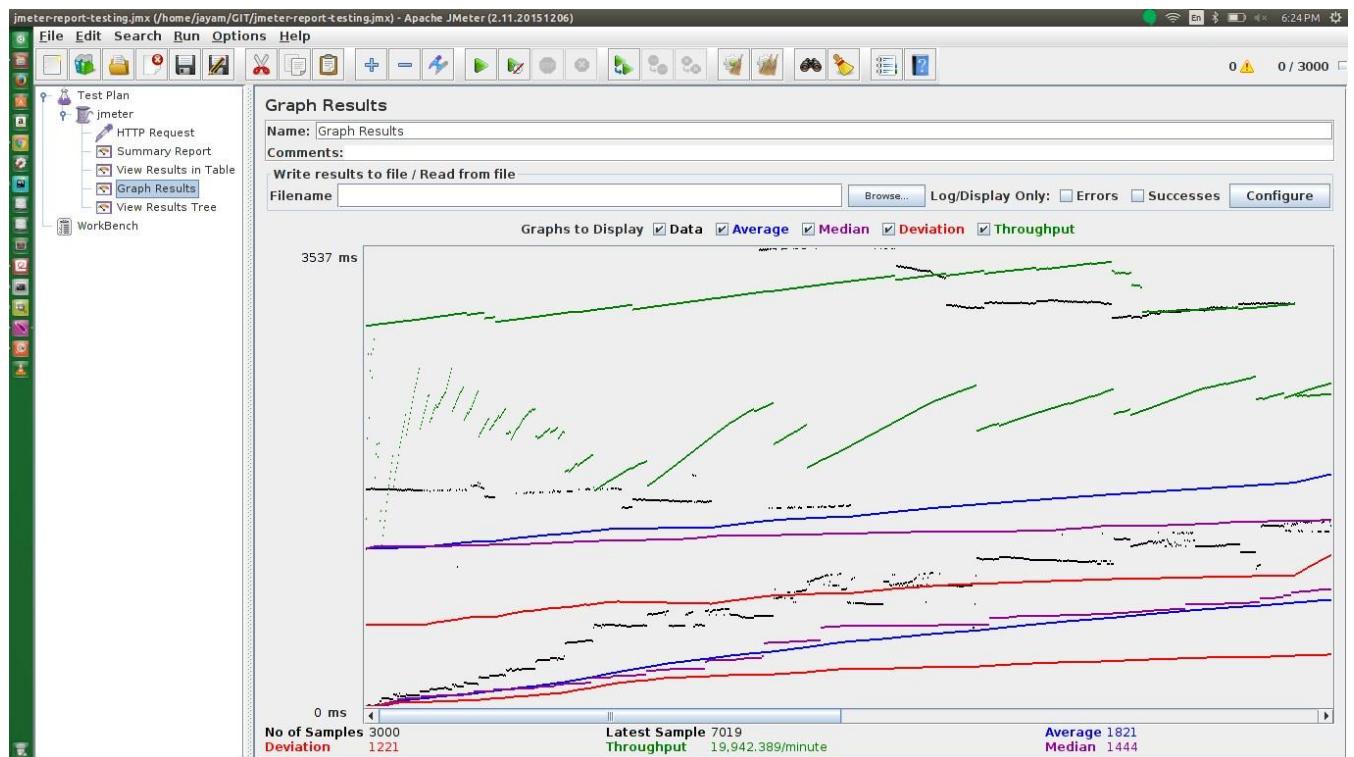
- **10000 Requests** - Base / Connection minute - Pooling
- Throughput - 6736
- Average 16178.176



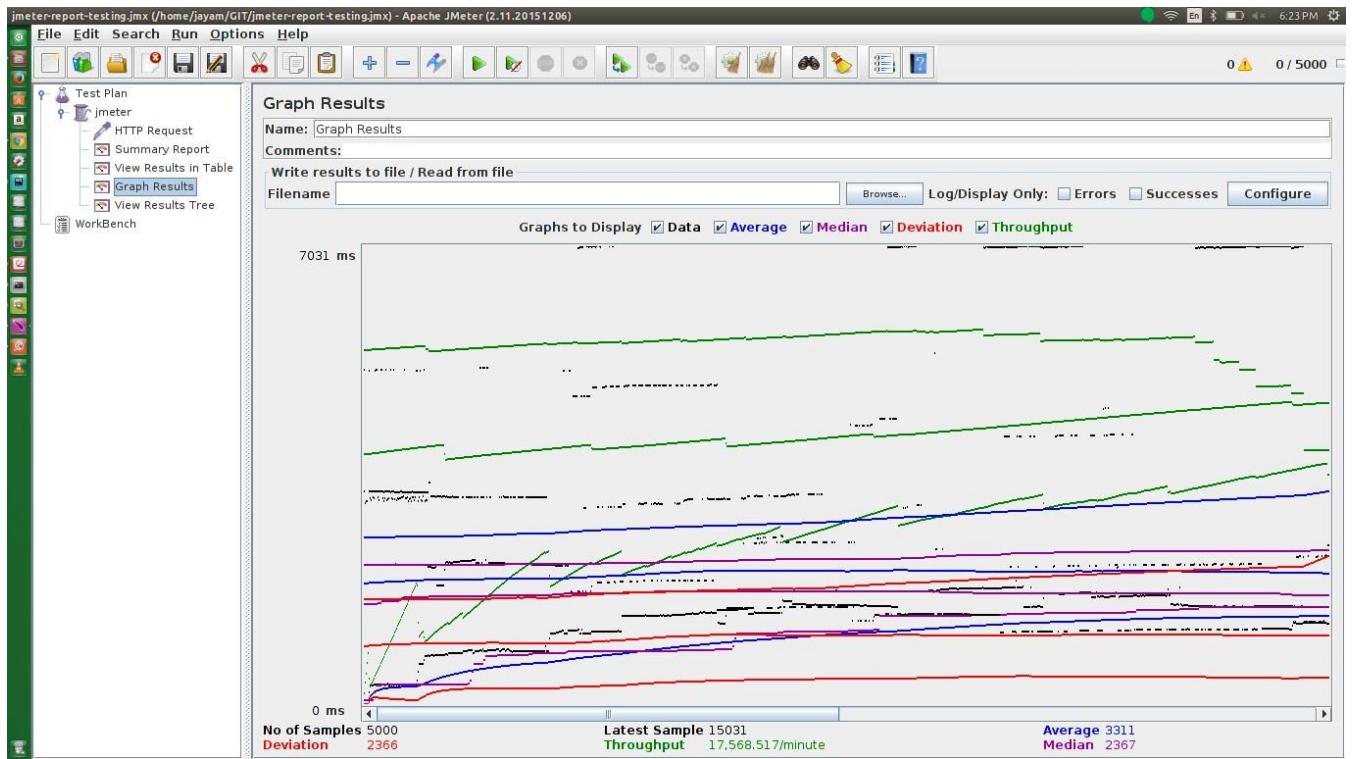
- **1000 Requests** - Base + Connection Pooling +  
 RabbitMQ Throughput / minute- 24135.157  
 Average - 471



• <b>3000 Requests</b>	-	Base	+	Connection	Pooling	+
RabbitMQ	Throughput	/		minute-	19942.389	
Average	-	1821				

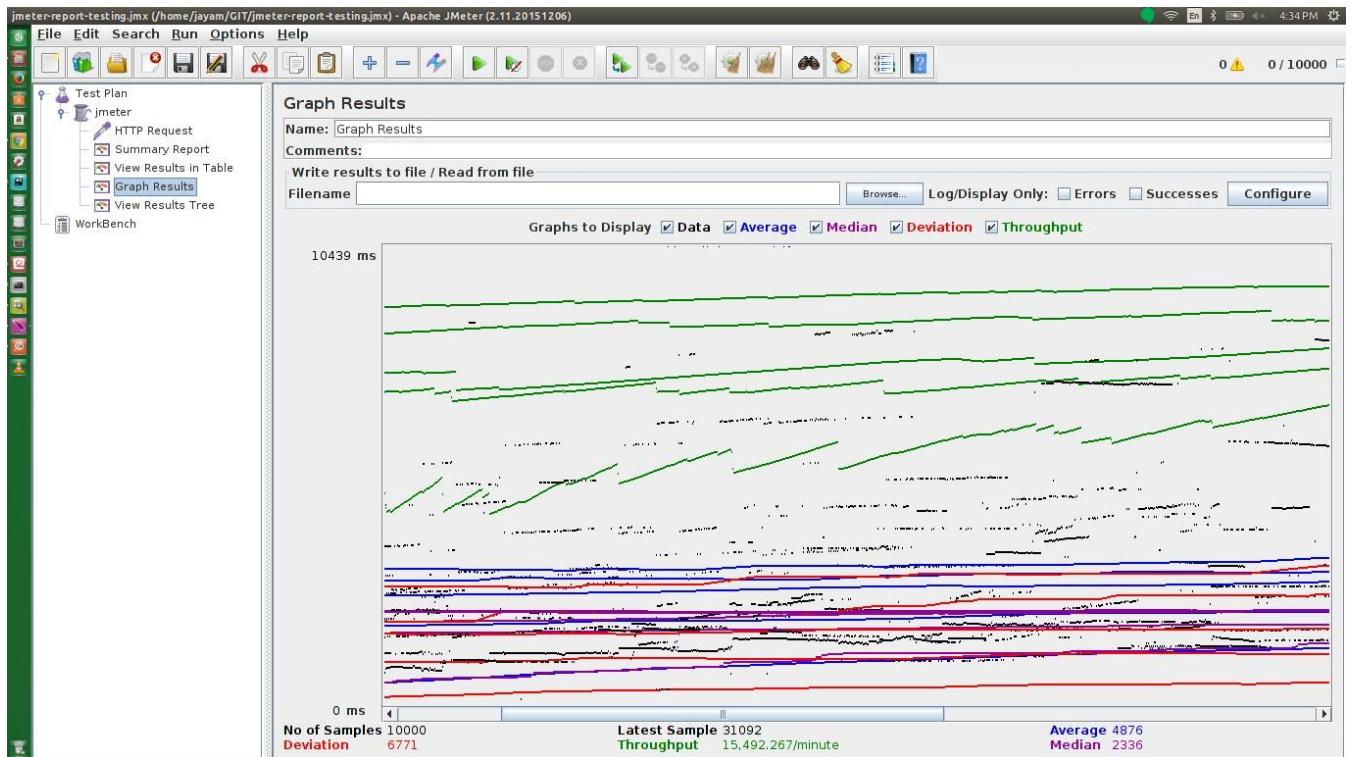


• <b>5000 Requests</b>	-	Base	+	Connection	Pooling	+
RabbitMQ	Throughput	/		minute-	17568.517	
Average	-	3311				

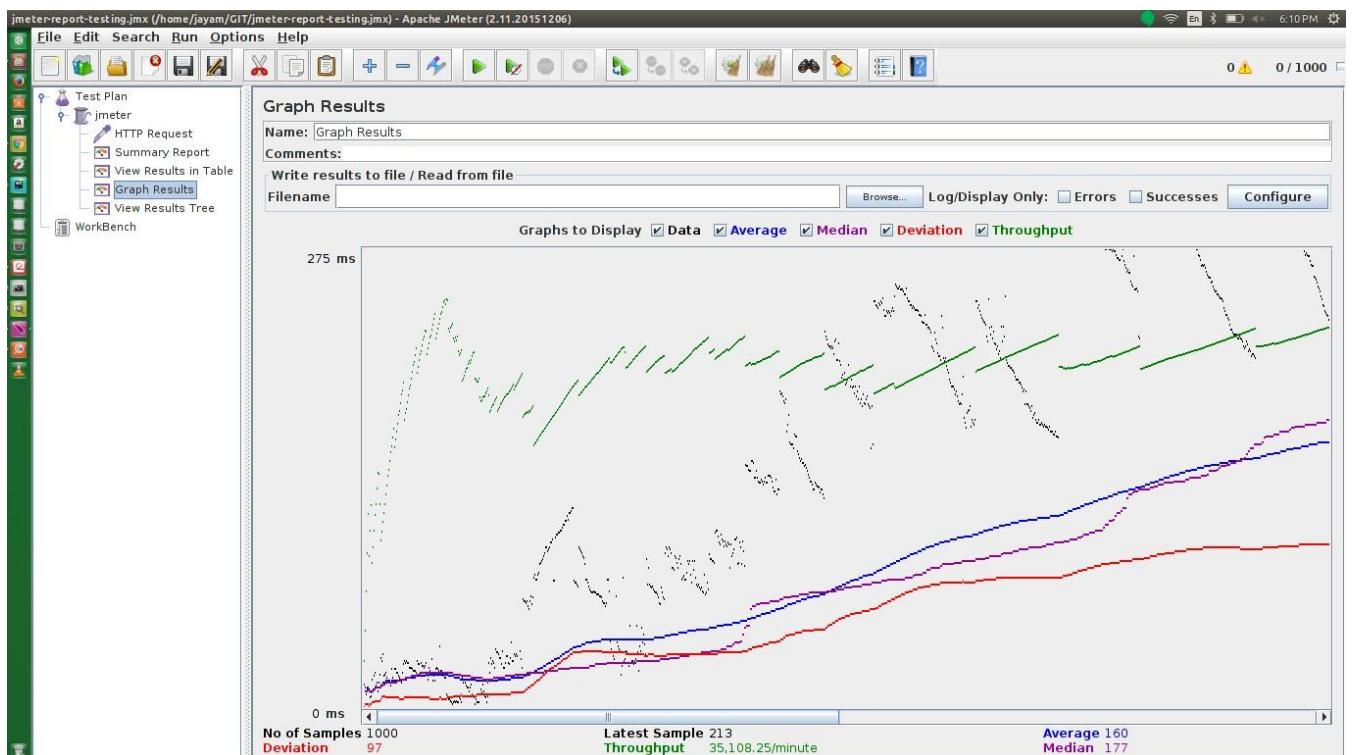


- **10000 Requests** - Base + Connection Pooling +
 

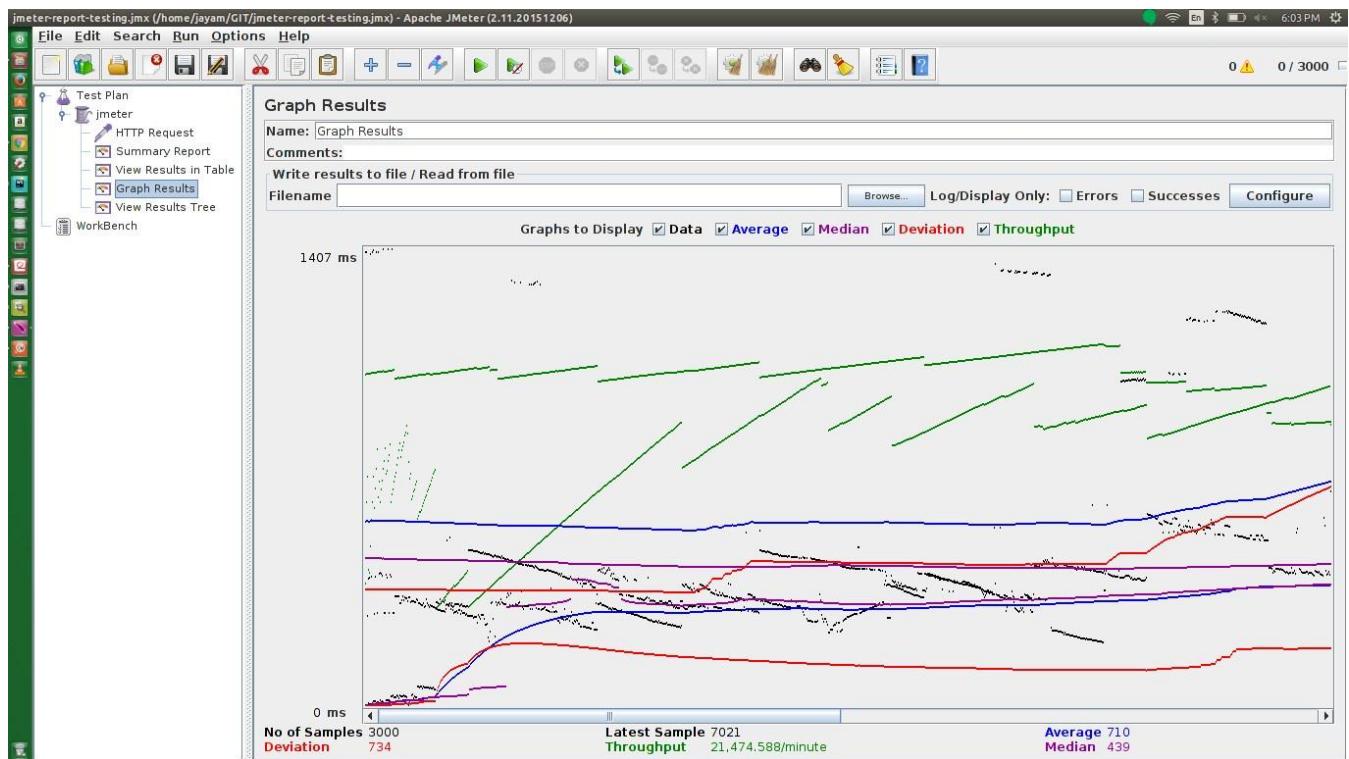
RabbitMQ	Throughput / minute	15492.267
Average	- 4876	



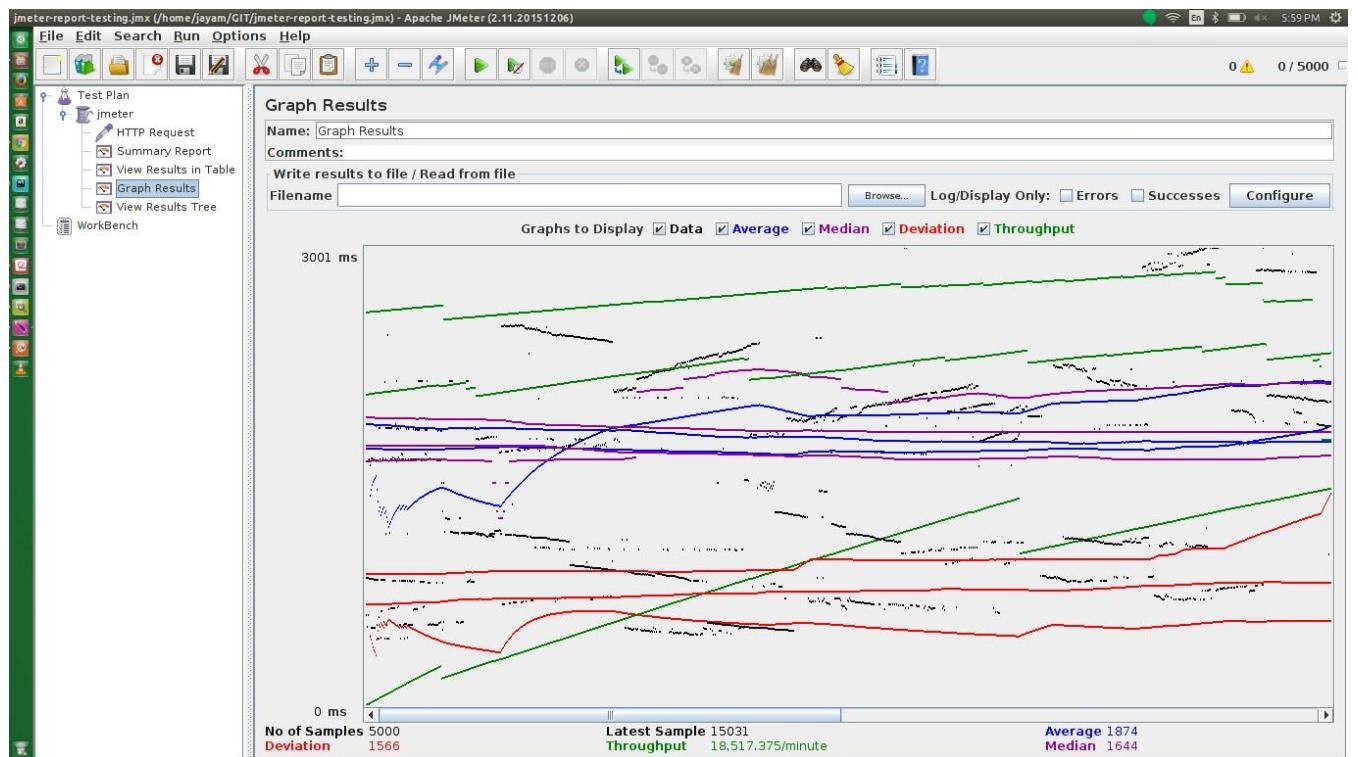
- **1000 Requests** - Base + Connection Pooling +
- RabbitMQ + Redis
- Throughput / minute - 35108.25
- Average - 160



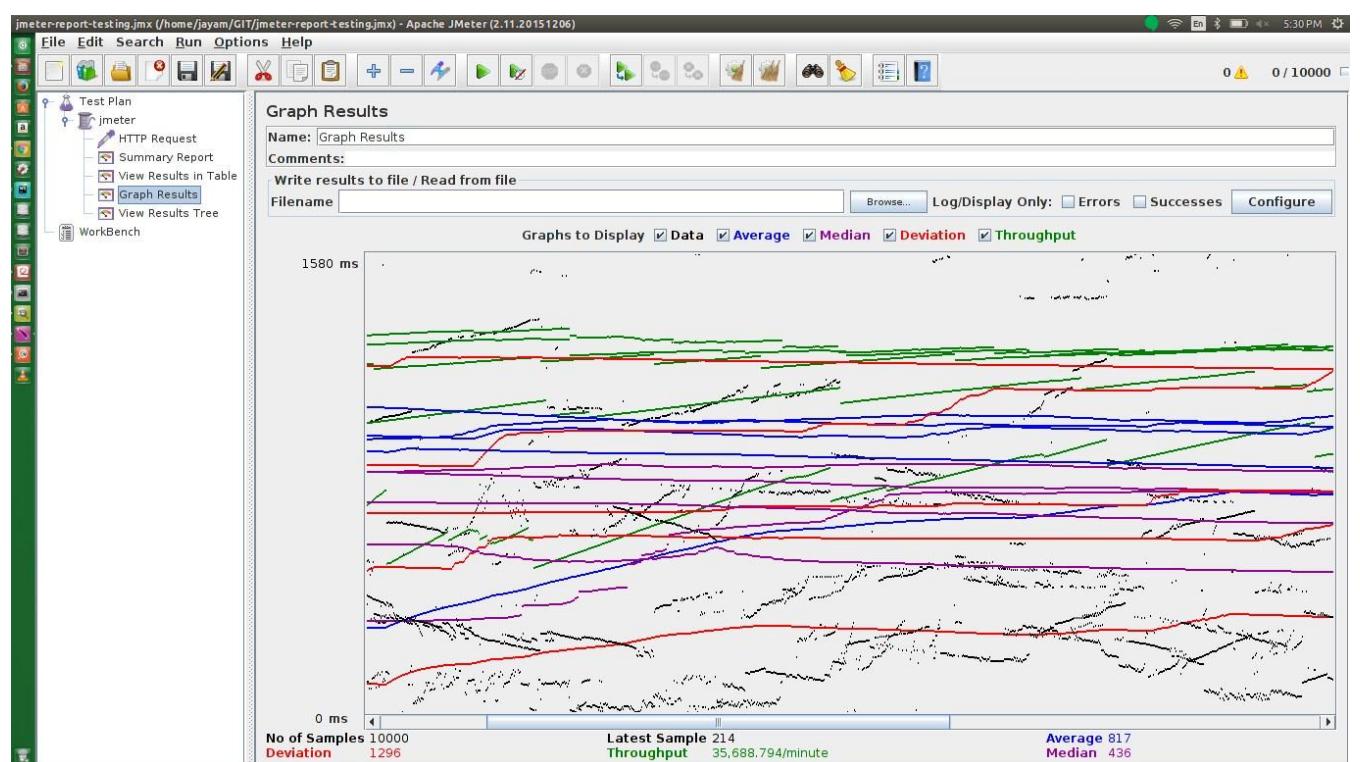
•	<b>3000 Requests</b>	-	Base	+	Connection Pooling	+	
	RabbitMQ	+	Redis				
	Throughput		/		minute	-	21474.588
-			710				Average



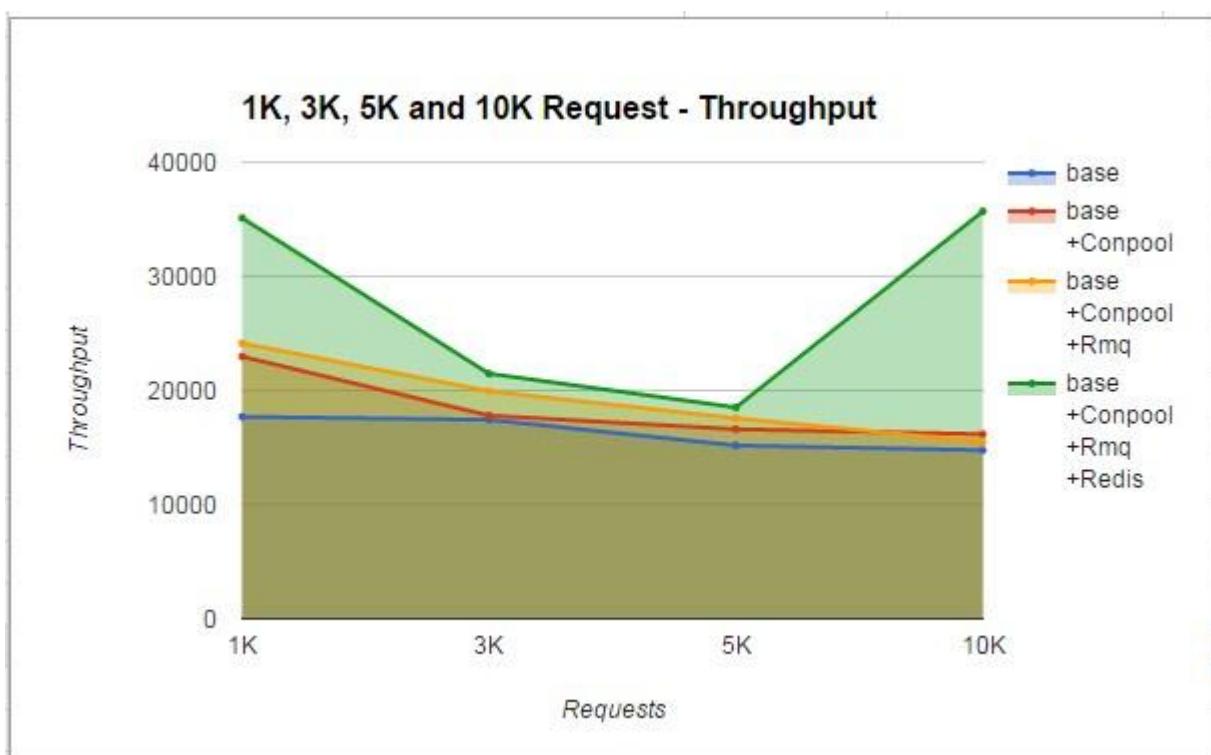
•	<b>5000 Requests</b>	-	Base	+	Connection Pooling	+
	RabbitMQ	+	Redis			
	Throughput		/	minute	-	18517.375
		-				Average
				1874		



• <b>10000 Requests</b>	-	Base	+	Connection	Pooling	+
RabbitMQ	+	Redis				
Throughput		/		minute	-	35688.794
	-					Average 817

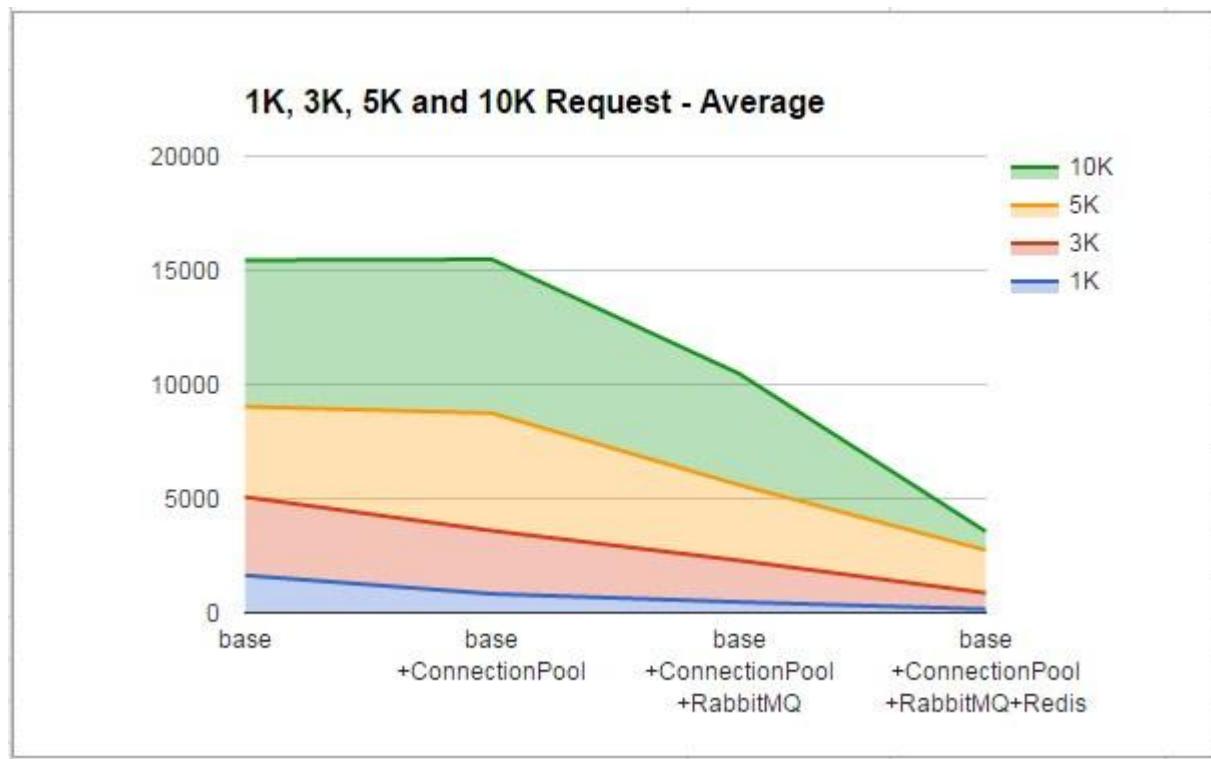


❖ Comparison of Throughput gain after adding features



As we can observe from the above graph, we have improved our applications overall throughput after adding performance gaining features like database connection pooling, RaabitMQ and Redis. Among these we have found to gain maximum throughput when using RabbitMQ and Redis in conjunction. RabbitMQ gives us ability to handle huge number of requests by pushing them in to asynchronous message queues, this gives us ability to absorb huge traffic of incoming requests and still respond to each without losing any request. On top of this we have used Redis, Redis gives us this unique ability to cache data in memory. Redis is an in memory database, this makes it an ideal candidate to cache resources effectively. This is a great advantage for huge traffic and load bearing technologies working in harmony with each other, we have been able to achieve this performance.

#### **Comparison of Average time after adding features**



## ❖ Mocha Testing

```
cmd C:\Windows\system32\cmd.exe
fixedPrice: 123,
title: 'Paradise Centre',
description: 'Beautiful Location',
toDate: '2017-01-31T08:00:00.000Z',
fromDate: '2016-11-30T08:00:00.000Z',
auctionPrice: null,
city: 'San Jose',
longitude: '-121.8944445',
latitude: '37.3403981',
listId: '769-40-8913',
status: 'rejected',
hostId: 'katy@gmail.com',
__v: 0,
review: [],
created_at: '2016-11-25T08:00:00.000Z',
listingImages: [ [Object] ],
facilities: [ 'Laundry - dryer' ],
safetyAmmneties: [ 'WifSmoke Detector' ],
basicAmmneties: [ 'TV', 'Heat' ],
address: [ [Object] ] > ]
Got all the rejected listings
POST /adminGetAllRejectedListings 200 10.515 ms - 988
[ { _id: '583b28946f6cab8e4ac14518',
  hostName: 'Katy',
  hostProfilePic: 'https://s3.amazonaws.com/cmpe273airbnb/Katy.jpg147985544888
9',
  guestAllowed: 2,
  bathCount: 1,
  roomType: 'Entire Place',
  fixedPrice: 123,
  title: 'Paradise Centre',
  description: 'Beautiful Location',
  toDate: '2017-01-31T08:00:00.000Z',
  fromDate: '2016-11-30T08:00:00.000Z',
  auctionPrice: null,
  city: 'San Jose',
  longitude: '-121.8944445',
  latitude: '37.3403981',
  listId: '769-40-8913',
  status: 'rejected',
  hostId: 'katy@gmail.com',
  __v: 0,
  review: [],
  created_at: '2016-11-25T08:00:00.000Z',
  listingImages: [ [Object] ],
  facilities: [ 'Laundry - dryer' ],
  safetyAmmneties: [ 'WifSmoke Detector' ],
  basicAmmneties: [ 'TV', 'Heat' ],
  address: [ [Object] ] } ]
✓ Admin: should get all rejected listings

10 passing (2s)
Clean Redis
C:\Projects\273 Project 20161204\--AirBnb--\airbnb-client\node_modules\.bin>
```

## **Dynamic Pricing:**

**Dynamic Pricing -** Dynamic pricing will be applied using four parameters.

- 1: Property Clicks**
- 2: User Reviews**
- 3: User Rating**
- 4: Property Rating**
- 5: Festival Days Special Offer**

Our first most prominent parameter for doing dynamic pricing for a property will be the property popularity itself, the property that has more number of clicks will be considered will have more impact for dynamic pricing. On top of that we will consider the user reviews for the property, the property that is more user favourable will be more costlier as compared to rest. After that we will consider user rating of the user having listing, if the user has more rating than his property will have more affinity for costlier price & vice versa. After that we will consider the property rating, the property having more ratings will be costlier & vice versa. After all these parameters we will also consider the Festival days, these will be the final dynamic part. We will apply this model for dynamic pricing.

❖ Ionic Screenshots



Welcome to Airbnb

Conitnue with facebook

Create Account

Log In



Welcome to Airbnb

Conitnue with facebook

Create Account

Log In

Android

The screenshot shows a mobile application interface for a travel or vacation booking platform. At the top, there is a navigation bar with a back arrow icon on the left and four tabs on the right: 'HOME' (highlighted with a blue underline), 'TRIPS', 'HOST', and 'PROFILE'. Below the navigation bar is a red header bar with the text 'Featured Destinations' in white. The main content area displays four destination cards, each featuring a landscape image of the Eiffel Tower at sunset and the name of the destination overlaid in white text. The destinations are arranged in a 2x2 grid:

- Top-left: San Jose
- Top-right: Los Angeles
- Bottom-left: London
- Bottom-right: Beverly Hills

# Lets's Signup

Firstname

Lastname

Email

Password

Login

# Log In

Invalid Password

mak@gmail.com

.....

---

Login

# Log In

Email id not found, Please register

makasdsad@gmail.com

---

.....

---

Login

Check In date      Check Out date      Guest

mm/dd/yyyy      mm/dd/yyyy     

---

Room type



Entire Home



Private Room



Shared Room

Rentals •





Entire Home



Private Room



Shared Room

Filter

Rentals •



\$123 ⚡

Harry Potter Wizard

Entire Place • 1 guests •





## Harry Potter Wizard

London, GB, England • ★★★★☆ 10 reviews



Shaktiman





### 100% refundable

Cancel up to 30 days before your trip and get a full refund, including service fees.

## About this listing

Wizardly Magically true

Accommodates: **1**

Available From: **2016-12-09**

Bathrooms: **1**

Available Till:

**2017-03-31**

Bedrooms: **2**

Property type: **Apartment**

Beds: **2**

Room type: **Entire Place**

### Amenities

Essentials

Wifi

Shampoo

Closet/drawers

Tv

linen

TV

Heat

Air Conditioning

Smoke Detector

Air Conditioning

First Aid and Kit

### Prices

Extra people: **\$123 / night**

**after the first day**

**\$123**

Check In

mm/dd/yyyy

Check Out

mm/dd/yy

Guests

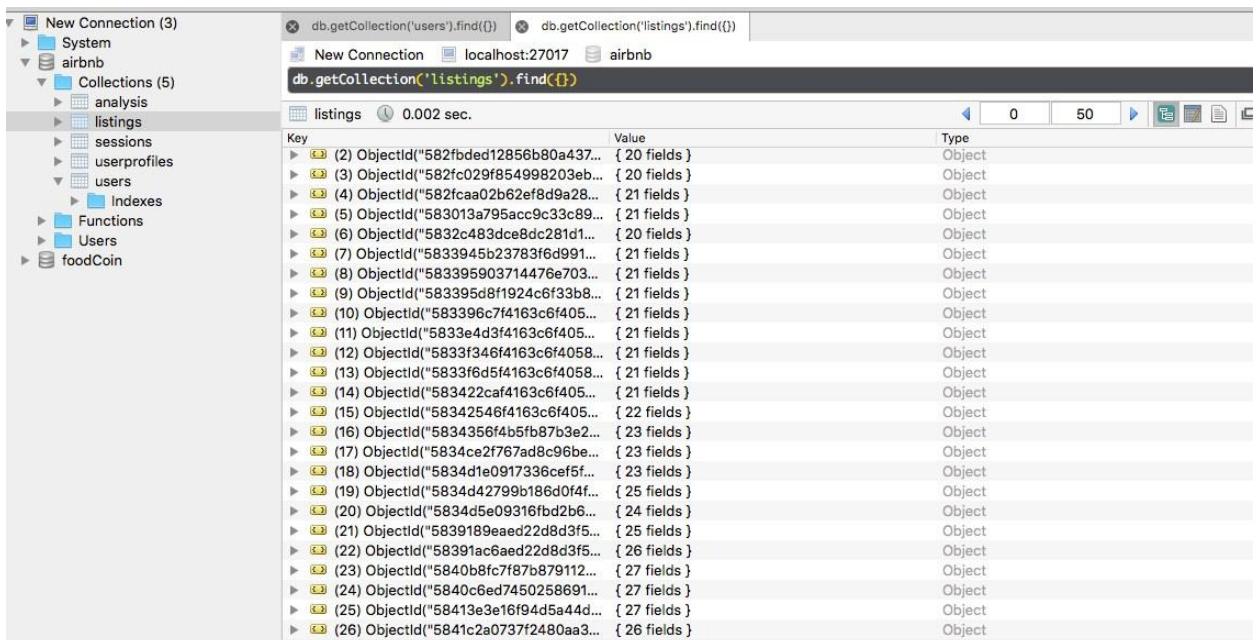


**Book**

You would not be charged yet.

## Code Listing:

### Database:



The screenshot shows the MongoDB Compass interface with the following details:

- Left Sidebar:** Shows the database structure with "New Connection (3)" selected. Under "airbnb", there are collections: "analysis", "listings", "sessions", "userprofiles", and "users".
- Top Bar:** Displays two tabs: "db.getCollection('users').find()" and "db.getCollection('listings').find()". The second tab is active.
- Result Grid:** The results of the "db.getCollection('listings').find()" query are displayed. The grid has columns: "Key", "Value", and "Type".
- Grid Data:** The "Value" column shows 26 documents, each represented by a yellow icon and a long ObjectId string. The "Type" column shows all entries as "Object".
- Bottom Navigation:** Includes buttons for "0", "50", and navigation arrows, along with icons for refresh, search, and export.

The screenshot shows the MySQL Workbench interface. On the left, there's a sidebar with various database management options like Server Status, Client Connections, and Data Import/Restore. Below that is the Object Explorer, which lists the 'airbnb' schema containing tables, views, stored procedures, and functions, and the 'sys' schema.

The main area has two tabs: 'Query 1' and 'SQL File 1'. The 'Query 1' tab contains the following SQL code:

```
1 • use airbnb;
2 • select * from trips;
3 • select * from trips where userEmail = 'shaktiman@gmail.com' AND
```

Below the code is a 'Result Grid' showing the results of the query. The columns are tripId, tripsequen..., userEmail, hostEmail, listId, and fixedPri. The data includes rows for various trips with different users and hosts.

tripId	tripsequen...	userEmail	hostEmail	listId	fixedPri
507-72-8526	9	katy@gmail.com	katy@gmail.com	444-48-5756	345
507-37-6021	10	katy@gmail.com	katy@gmail.com	530-81-1332	170
002-01-8602	11	shaktiman@gmail.com	krist@gmail.com	155-66-3862	245
044-60-7131	12	shaktiman@gmail.com	rock@gmail.com	003-32-9377	125
007-38-1824	13	katy@gmail.com	krist@gmail.com	155-66-3862	245
423-01-1031	14	katy@gmail.com	krist@gmail.com	155-66-3862	245
652-03-3156	15	krist@gmail.com	shaktiman@gmail.com	228-53-0418	123
750-03-2255	16	katy@gmail.com	katy@gmail.com	769-40-8913	123
NULL	NULL	NULL	NULL	NULL	NULL

At the bottom, there's an 'Action Output' section showing the execution history:

Time	Action	Response	Duration / Fetch Time
23:50:06	sel...	Error Code: 1046. No database selected Select the de...	0.058 sec
23:50:09	us...	0 row(s) affected	0.00042 sec
23:50:11	sel...	8 row(s) returned	0.00062 sec / 0.000...

Code Listing: Server Side:

```
www | scheduler.js | server.js
in the editListing_queue queue
in the isLoggedIn_queue queue
in the analysis_queue
in the getAllApprovedListings_queue queue
in the analysis_queue
in the getPaymentDetails_queue queue
in the getAllRejectedListings_queue queue
in the deleteAccount_queue
in the user_track
in the bidding_track_queue
in the analysis_queue
in the analysis_queue
in the getBillfromBillDate_queue queue
in the getUserfromCity_queue queue
in the getReviews_queue queue
in the total_revenue_queue
in the updateAuctionTrips_queue
in the placeBid_queue
in the getAllAuctionableProperties_queue
in the user_track
in the getAllProperties_queue
in the allCities_queue
in the total_listings_queue
in the checkDates_queue
in the getProfile_queue queue
in the bookTrip_queue
in the total_revenue_queue
in the getPendingTrips_queue queue
in the approveRequests_queue queue
in the editProfile_queue queue
in the paymentDetails_queue queue
in the pendingRequests_queue queue
in the approvedRequests_queue queue
in the completedTripsForUser_queue queue
in the rejectRequests_queue queue
in the reviewUser_queue queue
in the getAuctionTrips_queue queue
in the completedTripsForHost_queue queue
in the yourReservations_queue queue
```

Express

Session:

```

session| Match Case | Regex | Words | 14 matches
var uploadMyVideo;
//var imageProfileColl=[];

var mongoStore = require("connect-mongo")(expressSession);

var mongoSessionConnectURL = "mongodb://localhost:27017/airbnb";


var app = express();
var db = require('./models/db');
require('./routes/passport')(passport);

// view engine setup
app.set('views', path.join(__dirname, 'views'));
app.set('view engine', 'ejs');

//express-session

app.use(expressSession({
  secret: 'scope273_teststring',
  resave: false, //don't save session if unmodified
  saveUninitialized: false, // don't create session until something stored
  duration: 30 * 60 * 1000,
  activeDuration: 5 * 60 * 1000,
  store: new mongoStore({
    url: mongoSessionConnectURL
  })
}));

//express-session
// app.use(expressSession({



```

Code Listing for client side:

```

var mq_client = require('../rpc/client');

//for redis-----
var redis = require('redis');
var redisclient = redis.createClient(6379, '127.0.0.1');
//-----redis end--



function signin(req, res ,next)
{
  passport.authenticate('local-login', function(err, user, info) {
    console.log(" I am in login authenticate");
    if(err)
    {
      console.log("printing err EROB ERROR ERROR ERROR ERROR");
      if(err=="Timeout")
      {
        return res.send(err);
      }
      return next(err);
    }

    console.log("printing the info message",info);
    if(user && info.message=="notRegistered")
    {
      console.log("user not found notRegistered");
      return res.send({"msg":"404"});
    }

    if(user && info.message=="IncorrectPassword") {
      console.log("user not found IncorrectPassword");
      return res.send({"msg":"IncorrectPassword"});
    }
  });
}


```

## Database access classes:

```
1 var MongoClient = require('mongodb').MongoClient;
2 var db;
3 var connected = false;
4
5 exports.connect = function(url, callback){
6   MongoClient.connect(url, function(err, _db){
7     if (err) { throw new Error('Could not connect: ' +err); }
8     db = _db;
9     connected = true;
10    console.log(connected +" is connected?");
11    callback(db);
12  });
13};
14
15 exports.collection = function(name){
16   if (!connected) {
17     throw new Error('Must connect to Mongo before calling "collection"');
18   }
19   return db.collection(name);
20};
```

```
3 //Put your mysql configuration settings - user, password, database and port
4 function getConnection(){
5   var connection = mysql.createConnection({
6     host : 'localhost',
7     user : 'root',
8     password : 'q1d3m0',
9     //password: 'toor',
10    //database : 'airbnb',
11    password : 'system123',
12    database : 'airbnb',
13    port : 3306
14  });
15
16  return connection;
17}
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
```

function putData(callback,sqlQuery1,params){  
 console.log("\nSQL Query::"+sqlQuery1);  
 var connection=getConnection();  
 connection.query(sqlQuery1,params, function(err, rows, fields) {  
 if(err){  
 console.log("ERROR: " + err);  
 callback(err,null);  
 }  
 else{  
 console.log("Qty: " + rows);  
 callback(err,rows);  
 }  
 })  
}

```
    }
}

function yourListings(req,callback){
  console.log('In your listings queue' + req.username);
  Listing.find({hostId: req.username}, function(err, result){
    if(err){
      throw err;
    }else{
      console.log(result);
      if(result.length > 0){
        callback(null, {'code': '200', 'value': result});
      }else{
        callback(null, {'code': '201'});
      }
    }
  });
}

function deleteListing(req,callback){
  console.log('In delete listings queue' + req.id);
  Listing.update({listId: req.id}, { $set: {status: req.status} }, function(err, result){
    if(err){
      throw err;
    }else{
      if(result != ""){
        callback(null, {'code': '200'});
      }
    }
  });
}

function pendingRequests(req,callback){
  console.log('In pending requests queue' + req.email);
}
```

## Lessons Learned

The System has taught us in great detail on how the Enterprise solutions should be architected in order to create a large scalable and maintainable web application. We have created RESTful stateless service as a uniform communication API for requesting & sending resource. At the database level connection polling helped us in gaining better throughput by maintaining already created connections. We have also learnt that AMQP

protocol used in our application via RabbitMQ can be used as a asynchronous message passing mechanism. This gives us a great deal of scalability. On top of that applied with Redis, will make a web application even quicker by caching and sharing the resources from in memory cache.