CS 690: Graduate Project Portfolio

By

Geetika Koneru

Submitted in partial fulfillment of the requirements for the degree of

Master's in Applied Computer Science

In the Graduate College of

Southeast Missouri State University

Harrison College of Business and Computing

(August 2018 – December 2019)

Cape Girardeau, Missouri

3rd November 2019

Table of Contents Page No

Project 1: House price prediction and analysis based on logistic regression

- **Abstract:** This project aims at predicting the house prices applying python and machine learning skills. If we ask a home buyer for their dream house, they won't begin with the height of basement, proximity to east railroad and things of that sort. This way they are missing out many important features that aid in predicting the house price. This project explores 79 variables technically called features. Their importance in price prediction will be evaluated and exploratory data analysis will be performed to create new features thus making prediction of the sales prices in test dataset.
  Later, applying machine learning techniques, the model is trained to make accurate predictions.

- **Goal:** The Primary goal of the project is to make viable prediction on the sales price of houses in test dataset based on the training dataset.

- **Motivation:** The prime motivation to take up this project is to apply the learnt analytical principles in order to make analytical predictions thereby applying linear regression and Lasso principles using python libraries.

- **Problem Solution and contribution:** Handled the Machine Learning Regression part, Project idea and workflow development. To obtain accurate solution, analysis has been performed on the training data set where we performed exploratory data analysis, Data Cleaning, Plotting using varied types of maps, Data Aggregation and wrangled the features to create new viable features thereby eliminating the invalid, null and inappropriate one's that much less contributed for the feature analysis and prediction.

  - Algorithm: Linear Regression Lasso
  - Tools and Technologies: Jupyter Notebook, Python- NumPy, Matplotlib, Pandas
  - Deployment: This project is for analysis and the modules can be executed in a 6GB RAM intel core i5 processor.

- **Key results and Validation:** Applying the linear regression method and training the model using Lasso method, the sales price has been predicted and test scores for the model has been computed. Skew and Kurt values for the training and test datasets has been compared. The values have been observed to move towards normal distribution value of zero hence confirming that the prediction is a valid estimate of sales prices.

- **Keywords:** Lasso, Skew, Kurt, Linear Regression, Analytics and Prediction, Positive and Negative Correlation, Heat Map, RMSE(Root Mean Square Error).

Name of the Professor: Dr. Ziping Liu

Related Work and Procedure: The relative work included gathering of training and test datasets and setting up the environment to perform the data analysis workflow listed below:

- **Data Analysis:** The training data set has been analyzed to find the features that need to be enhanced, removed, aggregated and modified. The 79 features in the data set has been studied and identified the features that need modifications and removals.
- **Data Cleaning:** The identified features with null values and the one that much less contribute to the study has been eliminated from the dataset. The fields which give insights and have nulls at some places has been aggregated and filled with computed values.
- **Data Wrangling:** The data thus generated has been wrangled with NumPy and Pandas to generate new features that are combinational and can be used to predict the prices of houses in the test data set.
- **Data Aggregation:** The values computed are aggregated using the numerical and scientific computations.
- **Plotting:** Using Matplotlib, the data thus gathered is plotted across to determine the highly contributing features and valuable features thus aiding in the prediction of house prices.
- **Machine Learning Regression:** On the later stage with the refined training data set, Machine learning regression techniques have been applied namely Linear Regression Lasso. Feature identification has been performed using AVOVA test for disparity score calculation.
- We then applied the normalization, Root Mean Square Error(RMSE) for probabilistic analysis. From this, we categorized and separated the numerical and categorical variables. We could see from the results that there were 30 Numerical and 35 categorical data values and using AVOVA, we mapped variables to find the most contributing features. There was decreasing pval thus indicating the increasing diversity in partitions.
- Lasso Regression was used to train the model and the results were predicted for the test dataset.

Algorithm: Lasso Regression is a type of algorithm based on shrinkage where the values are shrunk towards the mean of the computational data.

Application Screenshots:

- Data Cleaning:

```
MiscFeature      1467 non-null object
MiscVal          1467 non-null int64
MoSold           1467 non-null int64
YrSold           1467 non-null int64
SaleType         1467 non-null object
SaleCondition    1467 non-null object
SalePrice        1467 non-null float64
dtypes: float64(5), int64(33), object(43)
memory usage: 928.4+ KB
```

SalePrice	Id
0.0	15
34900.0	1
35311.0	1

Here once we have appended the NaN values with Zero's we could see that SalesPrice has 1467 Non-null filled values for instance

```
In [35]:
unique_id = len(set(df.Id))
total_id = df.shape[0]
diff=total_id - unique_id
print("Number of duplicate Ids: " + str(diff))

Number of duplicate Ids: 7
```

```
In [36]:
df.drop_duplicates(inplace=True)
```

```
In [37]:
unique_id = len(set(df.Id))
total_id = df.shape[0]
diff=total_id - unique_id
print("Number of duplicate Ids: " + str(diff))

Number of duplicate Ids: 0
```

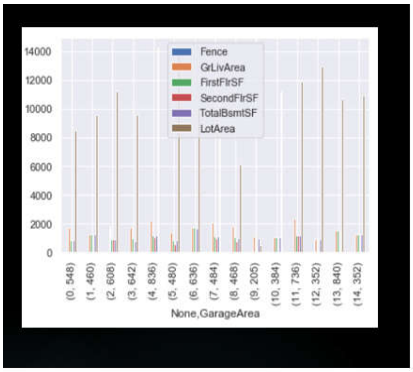
The duplicate rows of Id are dropped.

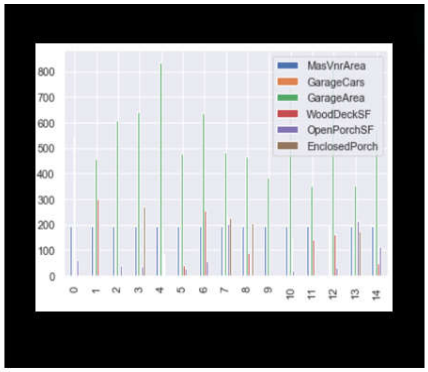
```
#Replacing missing data with 0
for col in ('GarageYrBlt', 'GarageArea', 'GarageCars','BsmtFinSF
    total_data[col] = total_data[col].fillna(0)
```

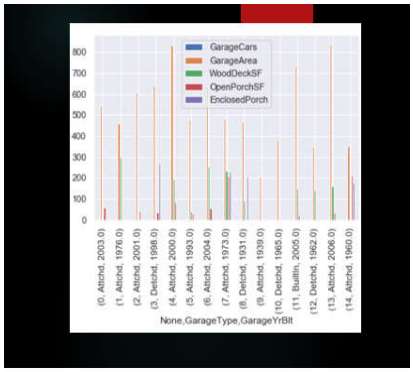
Replace None values with null values

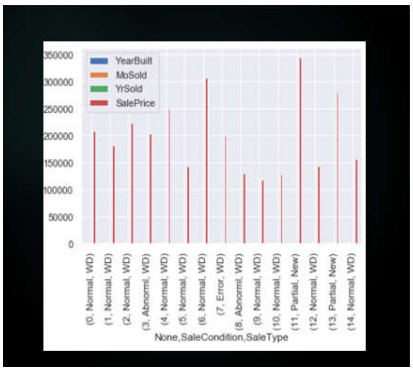
```
#Replacing missing data with None
for col in ['GarageType', 'GarageFinish', 'GarageQual', 'GarageC
    total_data[col] = total_data[col].fillna('None')
```

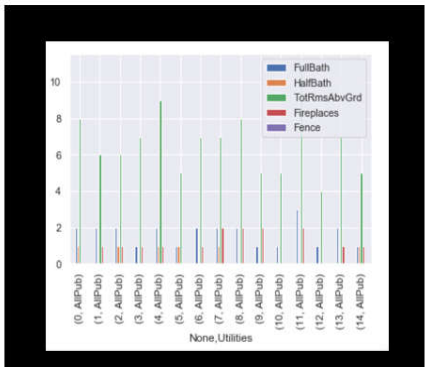
```
gargrp = ['GarageType', 'GarageFinish', 'GarageQual', 'GarageCon
total_data.groupby('GarageType')[gargrp].count()
```

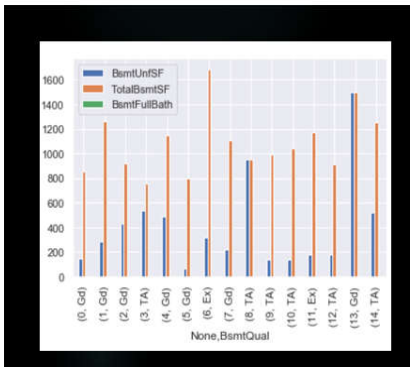




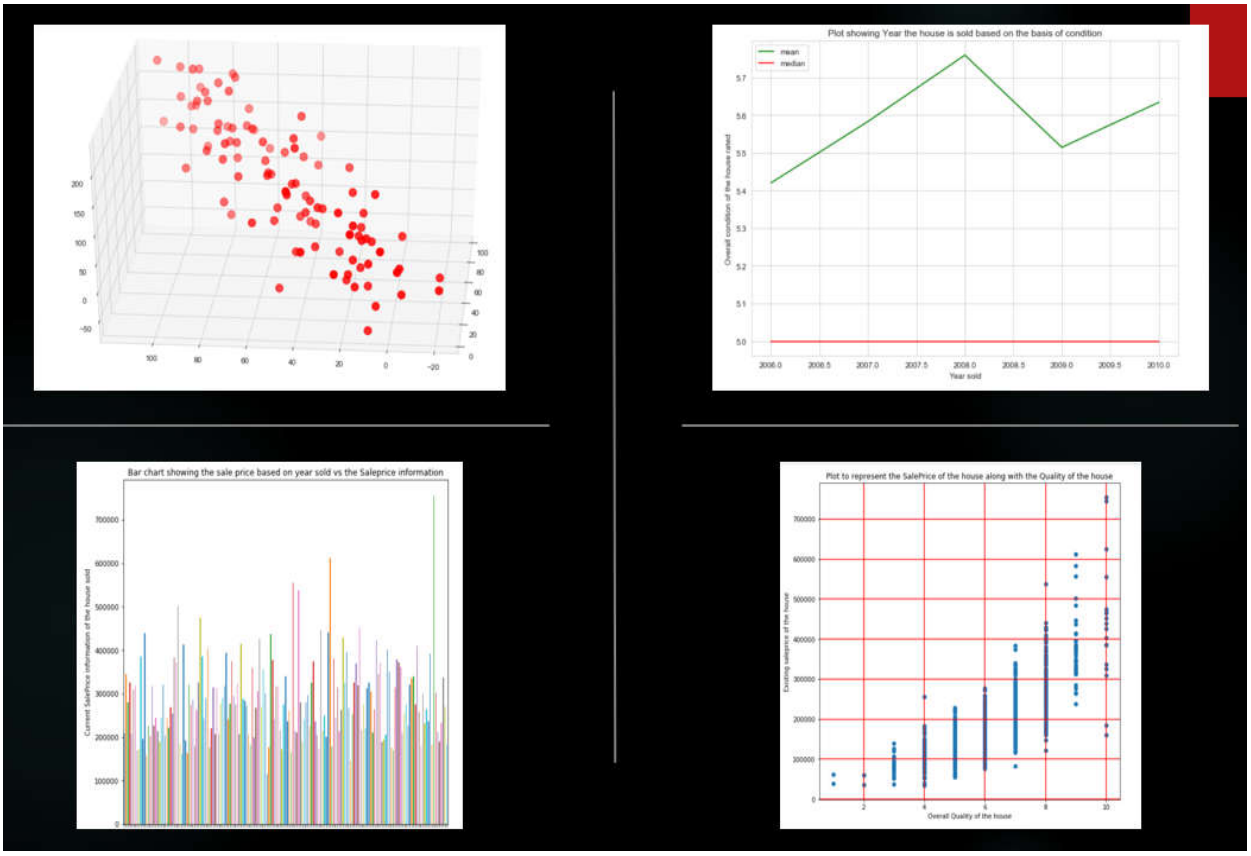


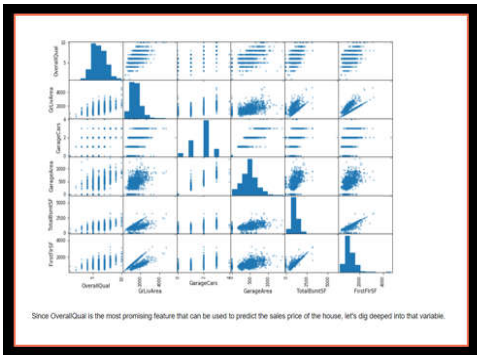


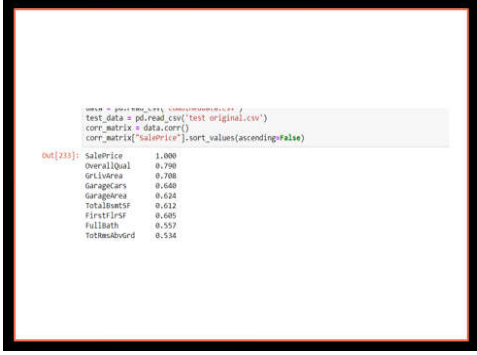




- Plotting:



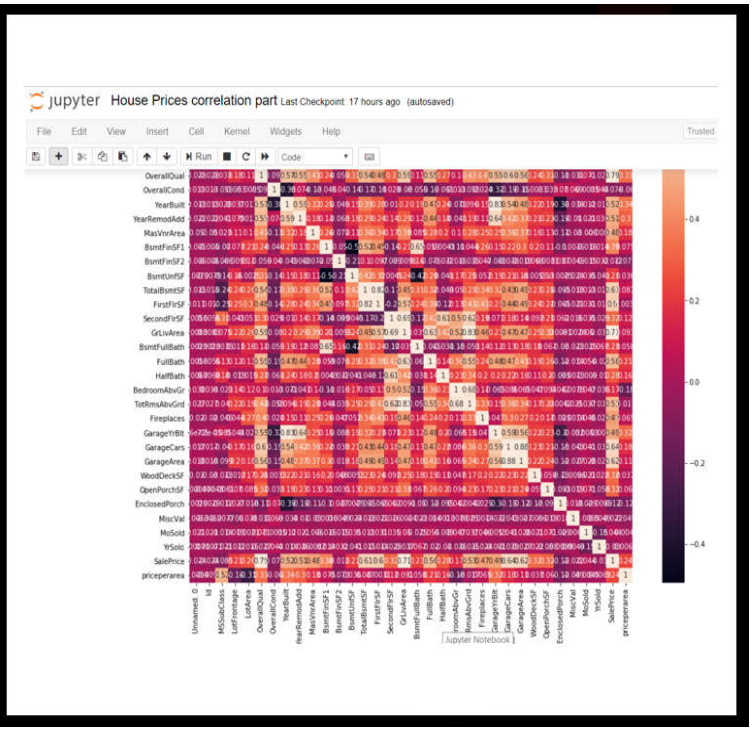




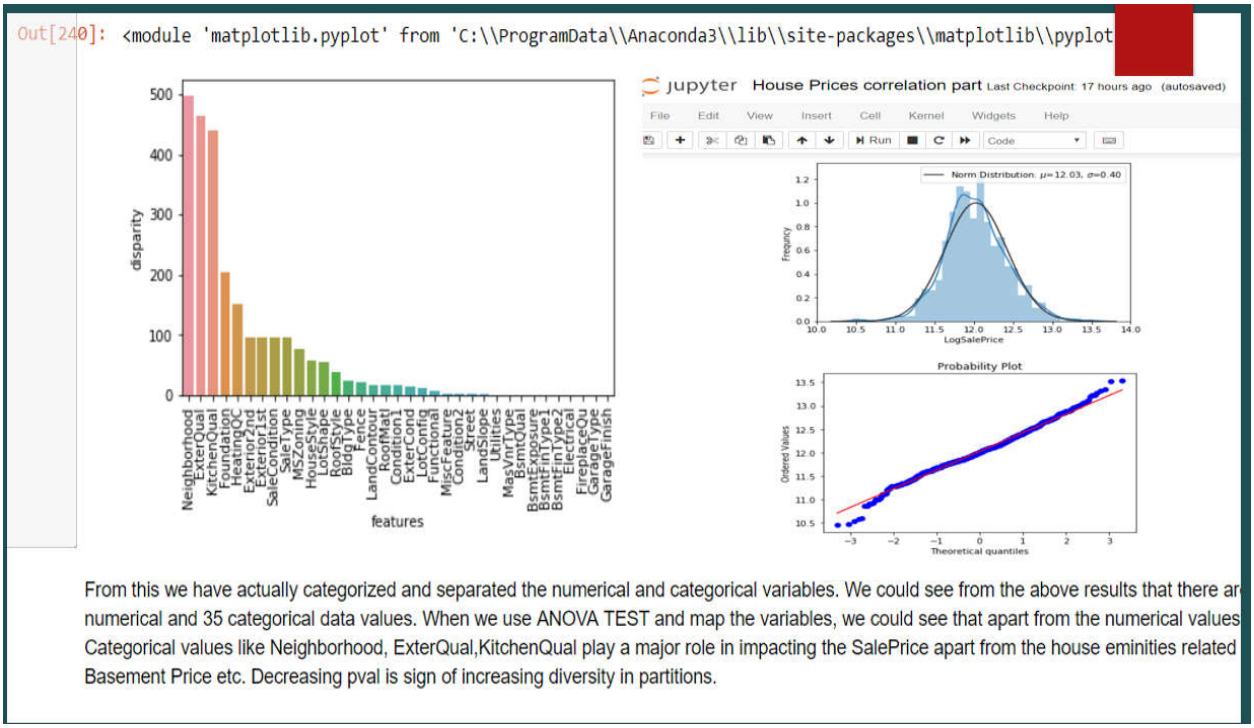
```
test_data = pd.read_csv('test original.csv')
corr_matrix = data.corr()
corr_matrix["SalePrice"].sort_values(ascending=False)
```

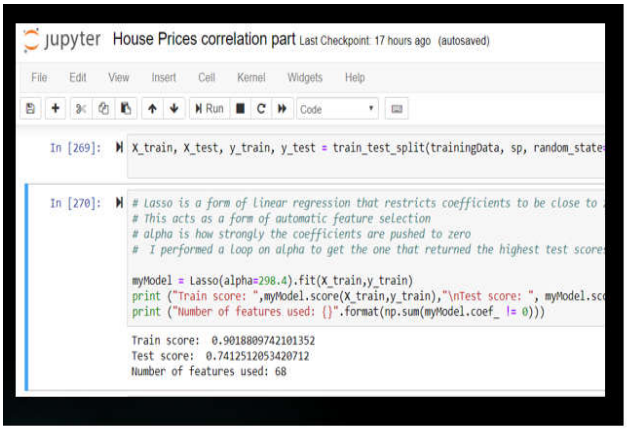
Out[233]:

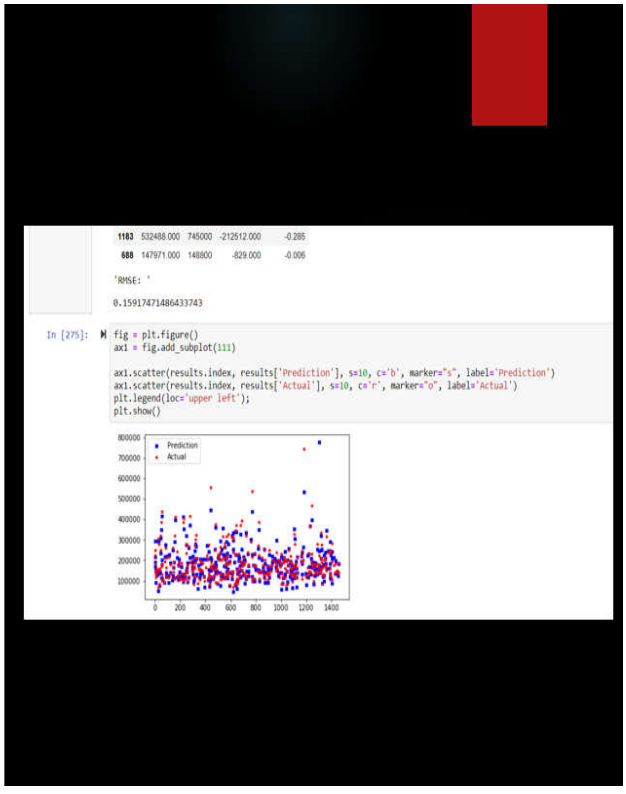
SalePrice	1.000
OverallQual	0.790
GrLivArea	0.708
GarageCars	0.640
GarageArea	0.624
TotalBsmtSF	0.612
FirstFlrSF	0.605
FullBath	0.557
TotRmsAbvGrd	0.534

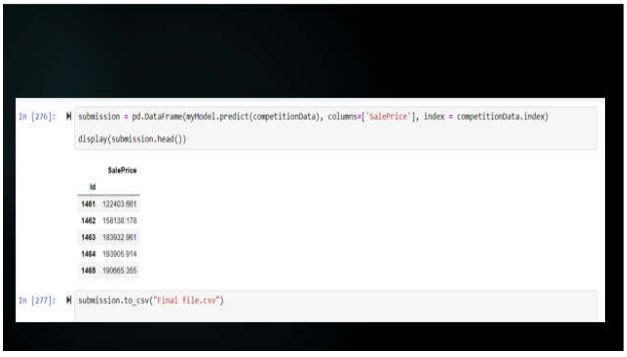


- AVOVA and Machine Learning with skew and Kurt results:









Project 2: Smart Devices Repair Handling System

Abstract:

The core idea of the project is to design a front-end responsive interface to fix the smart devices on time which might include onsite services and pickup options to make the users feel at ease. Access to the portal lets the users to choose services, tell about the type of device, track the progress and provide feedback for each individual service that they received.

Goal:

The goal of the project is to design a responsive front-end interface which is compatible to be viewed on laptops, mobile and handheld devices.

- Motivation: The key motivation to take up this project as part of web development course is to learn and apply JavaScript, Bootstrap, Angular and C# functionalities.

- **Contribution:** Project Idea, High Level Architecture Design, Bootstrap, Angular Implementation, Project and Team Management. Have turned the entire project to bootstrap design. Embedded page graphs. Demonstrated the bootstrap grid system in Home page. Designed an eye catchy interface.

  Currently, the project is in development and is yet to leverage the features of Angular implementation to improve speed and deploy application across all platforms.

- Tools and Technologies: Visual Studio Code, HTML 5, CSS3, JavaScript, Bootstrap 4, Angular 8, C#, ASP.NET
- Deployment: The application can be deployed using a simple browser and a laptop with core i5 processor or more with 6 GB RAM over all browsers.

- Key results and Validation: The outcome of the project resulted in a responsive website design that included validations implemented in JavaScript and interface in Bootstrap 4. The angular and backend development is still in progress till date. The navigation from one page to the another with respective alerts proves the application workflow.

- **Keywords:** Front-end, Responsive, Compatible, High level Architecture, Interface, Functionalities.

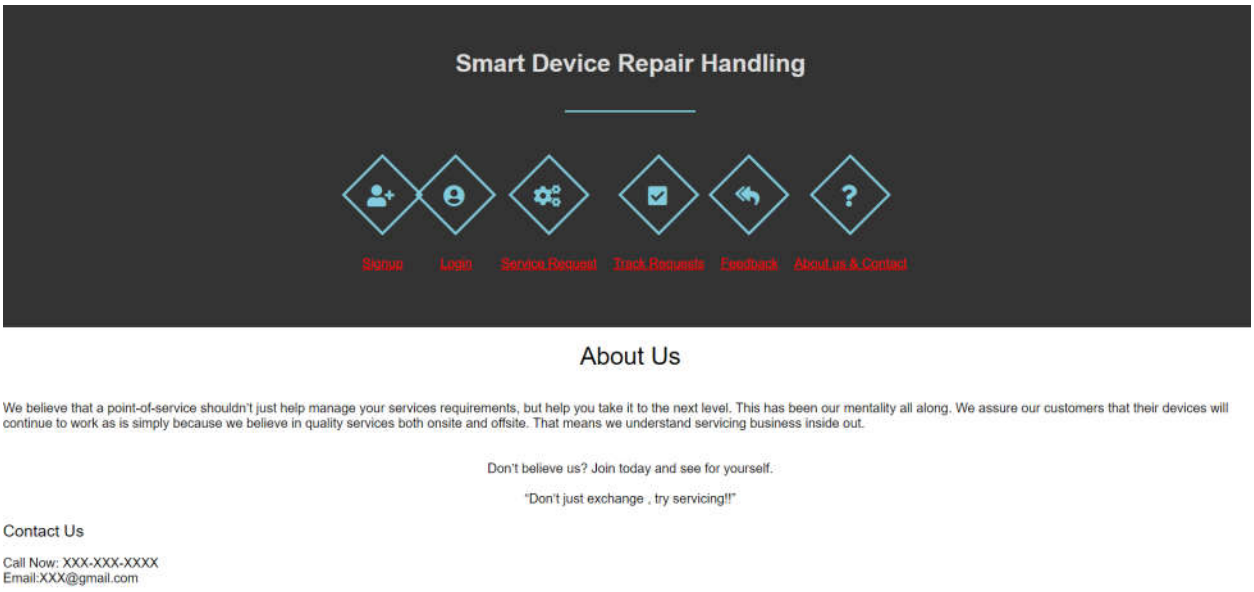
Name of the Professor: Dr. Ziping Liu

Related Work and Procedure: The application is a web-based platform to book smart devices repairing for home and onsite pickup. The application has been developed and has been made platform independent by adopting Bootstrap and Angular.

- The application involves full stack framework where the application development is done in stages. The first stage involved development of application site and its modules in HTML and CSS.
- Later in second stage, the application has been extended in Bootstrap and JavaScript. While in the third stage, the modules of the application have been broken down into Angular Components and added routing, services and HTTP services.
- We then have plans to connect it to the back-end database while writing the connections and scripts in C#.
- The modules involve service request page, tracking page, feedback page and Single page application home page compatible in both mobile and computers.
- We have designed everything from scratch and implemented the application in MVC(Model-View-Controller) architecture.

Application Screenshots:

- HTML and CSS pages:



Service Request Form

Name Of The Customer:

Age:

Contact:

Email:

Type Of Smart Device : select one option Mention If Other:

Select Service Type : Home Showroom Online Other

Mention If Other:

Select Appointment: mm/dd/yyyy

Comments:

Upload File

SUBMIT

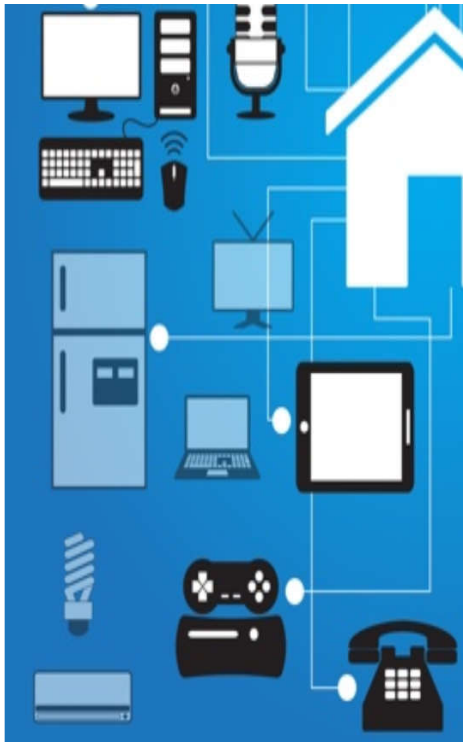
Tracking Information

Please enter your tracking Number:

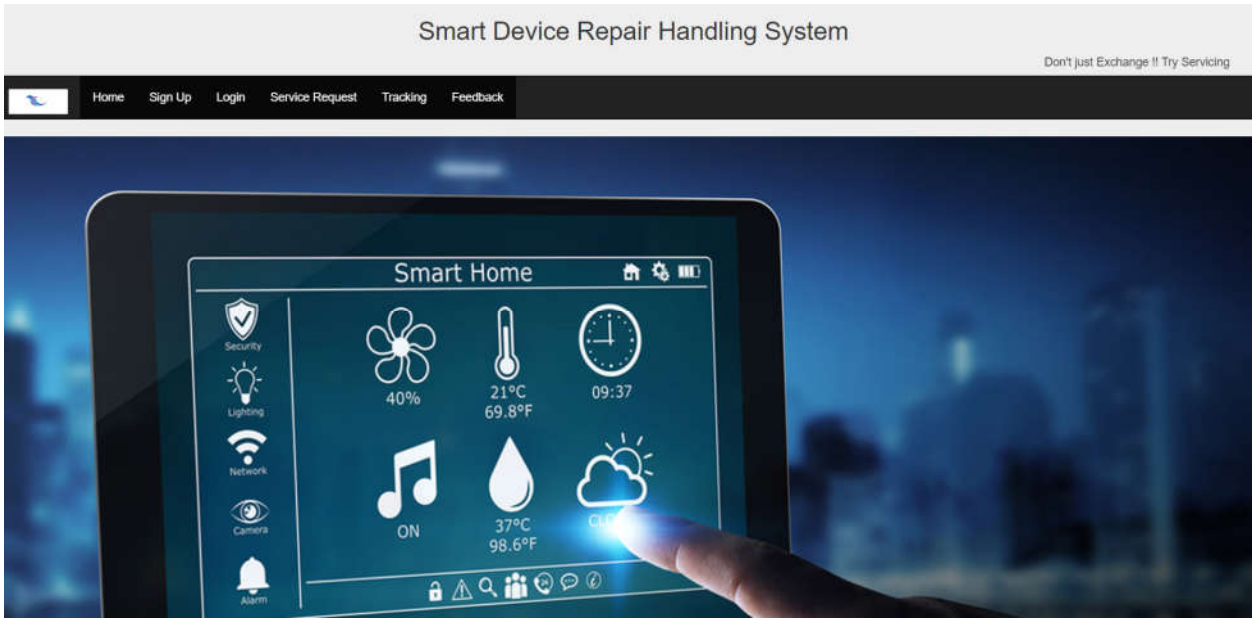
Submit

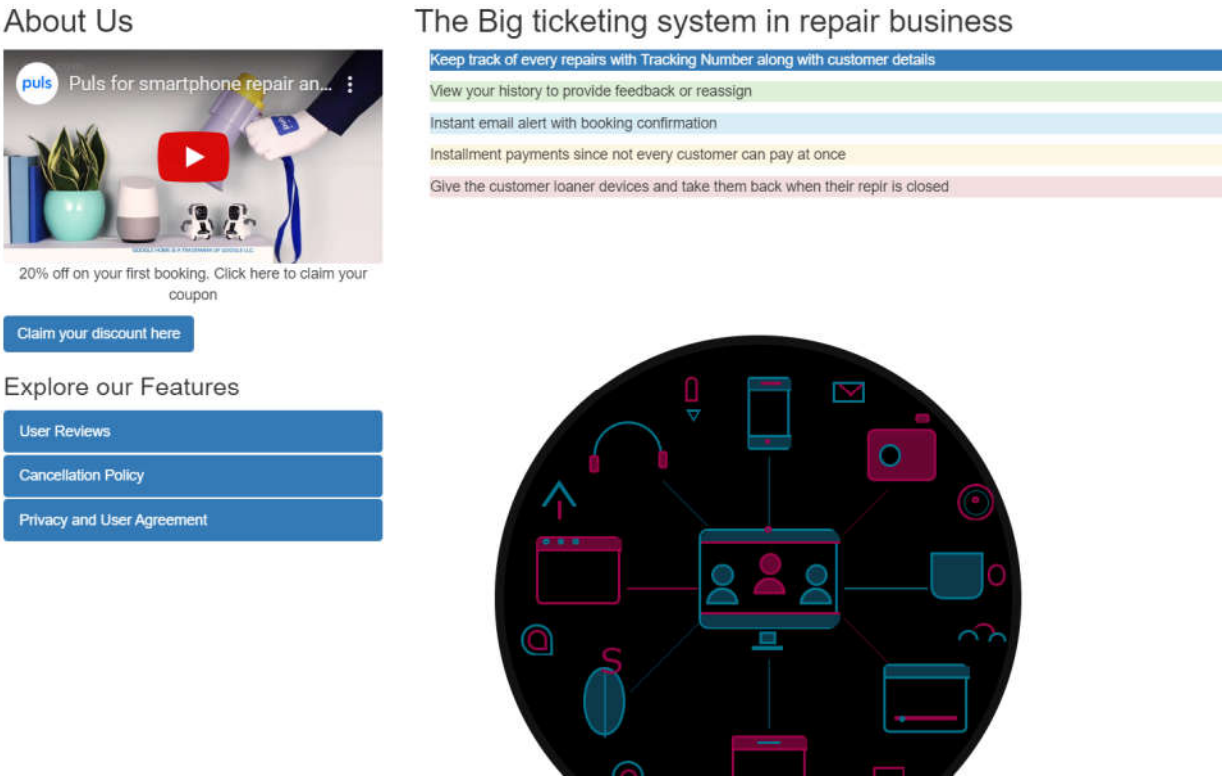
View Tracking history: From date: 08/31/2019 To date: 08/31/2019

Submit to view History

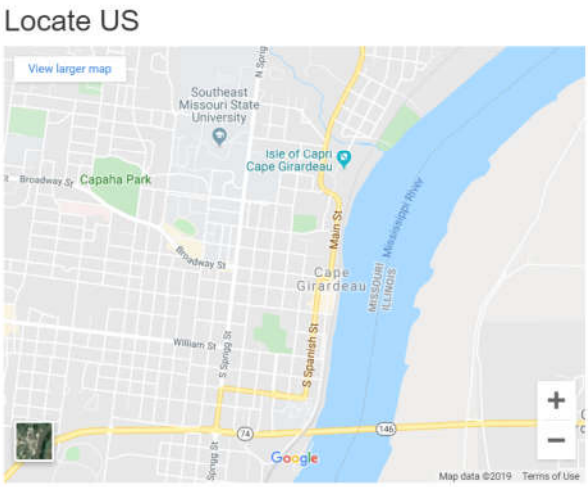


- Bootstrap:

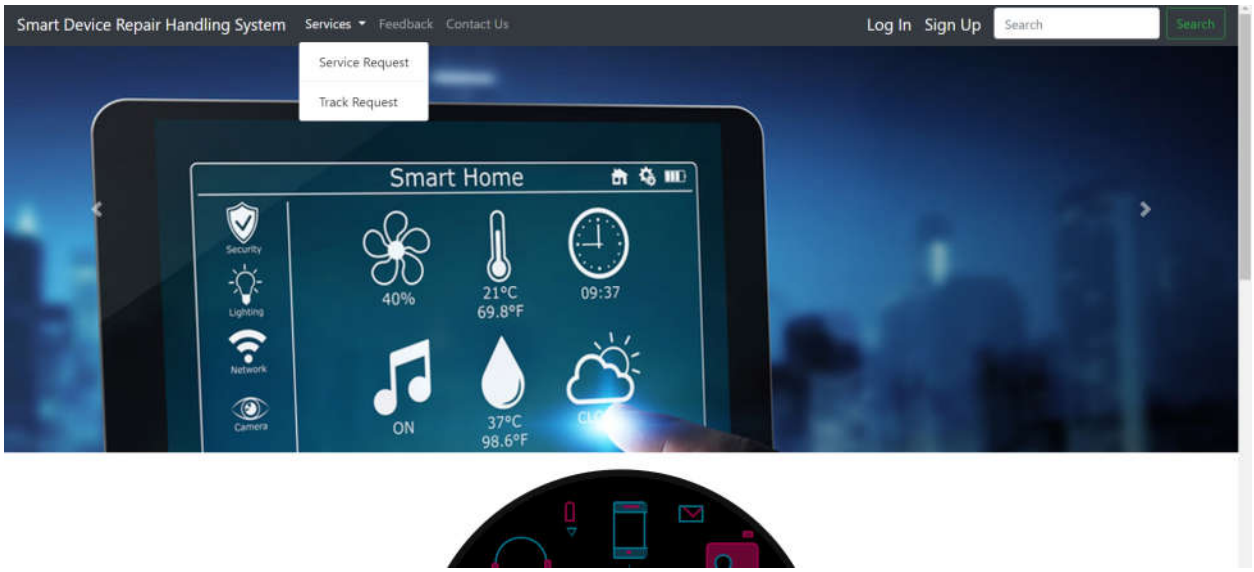

Smart Device Repair Handling System
Don't just Exchange !! Try Servicing
Home
Sign Up
Login
Service Request
Tracking
Feedback
Smart Home
Security
Lighting
Network
Camera
Alarm
40%
21°C
69.8°F
09:37
ON
37°C
98.6°F

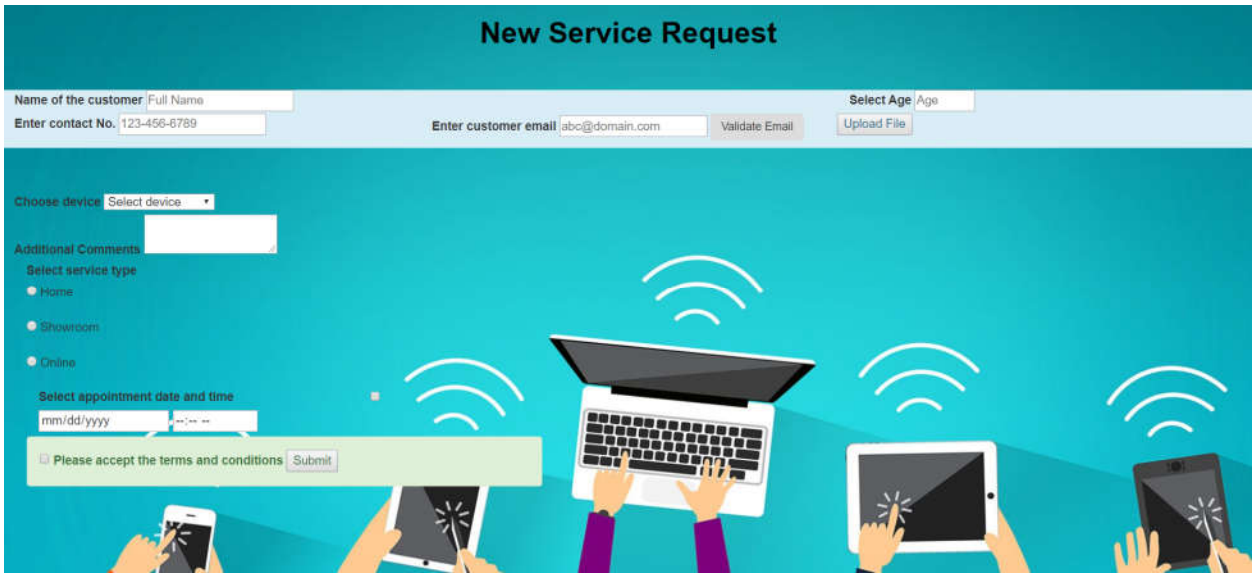

About Us
puls
Puls for smartphone repair an...
20% off on your first booking. Click here to claim your coupon
Claim your discount here
Explore our Features
User Reviews
Cancellation Policy
Privacy and User Agreement
The Big ticketing system in repair business
Keep track of every repairs with Tracking Number along with customer details
View your history to provide feedback or reassign
Instant email alert with booking confirmation
Installment payments since not every customer can pay at once
Give the customer loaner devices and take them back when their repir is closed


User Reviews
In home Service
My rating: ★★★★★☆
Never before has there been a good service. So why would not we service our devices? This latest attempt to bring service to our home. Thanks for excellent service and follow up ...
Raj from Srilanka
Im Impressed
My rating: ★★☆☆☆☆
It's been nice to have the website which asks us for choice. I got a chance to select my dealer. Happy to be served.
Mahesh United States
Can be improvized
My rating: ★★★★★★
Fantastic service
SriMukhi India

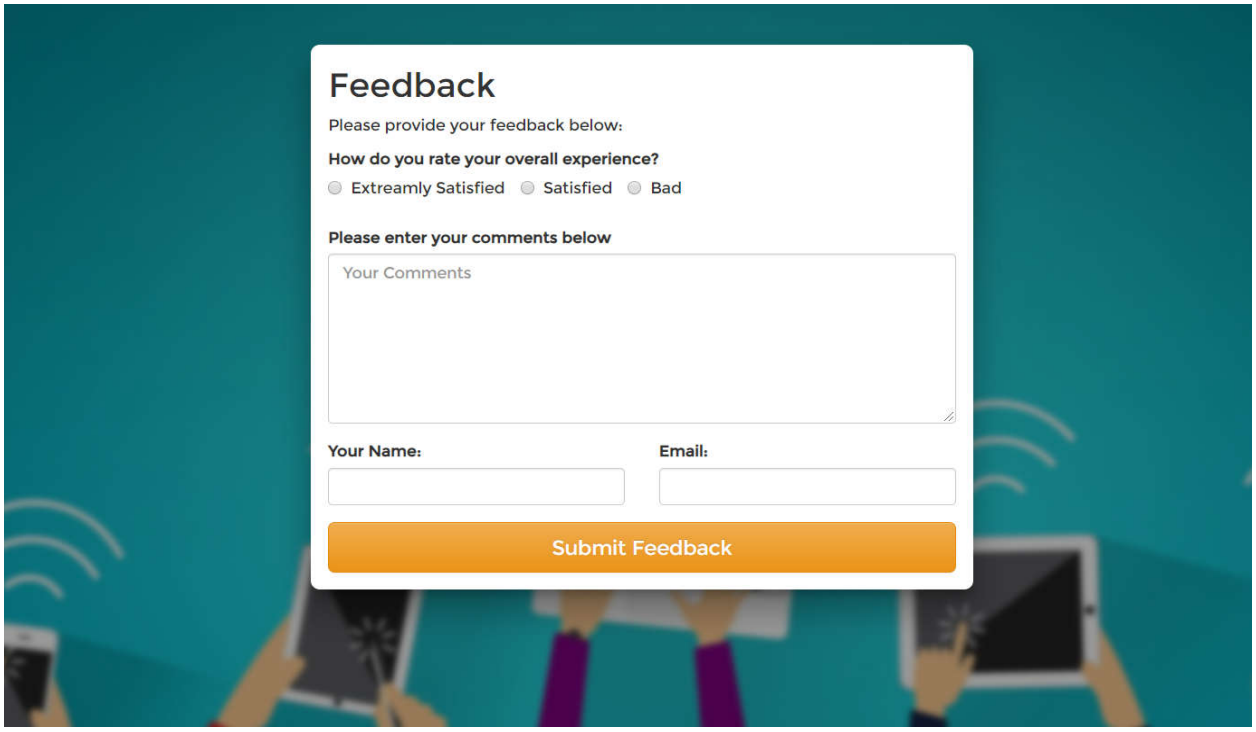

Locate US
View larger map
Southeast Missouri State University
Isle of Capri Cape Girardeau
Capaha Park
Broadway St
Main St
Cape Girardeau
Mississippi River
MISSOURI
ILLINOIS
William St
S Spanish St
Google
Map data ©2019 Terms of Use

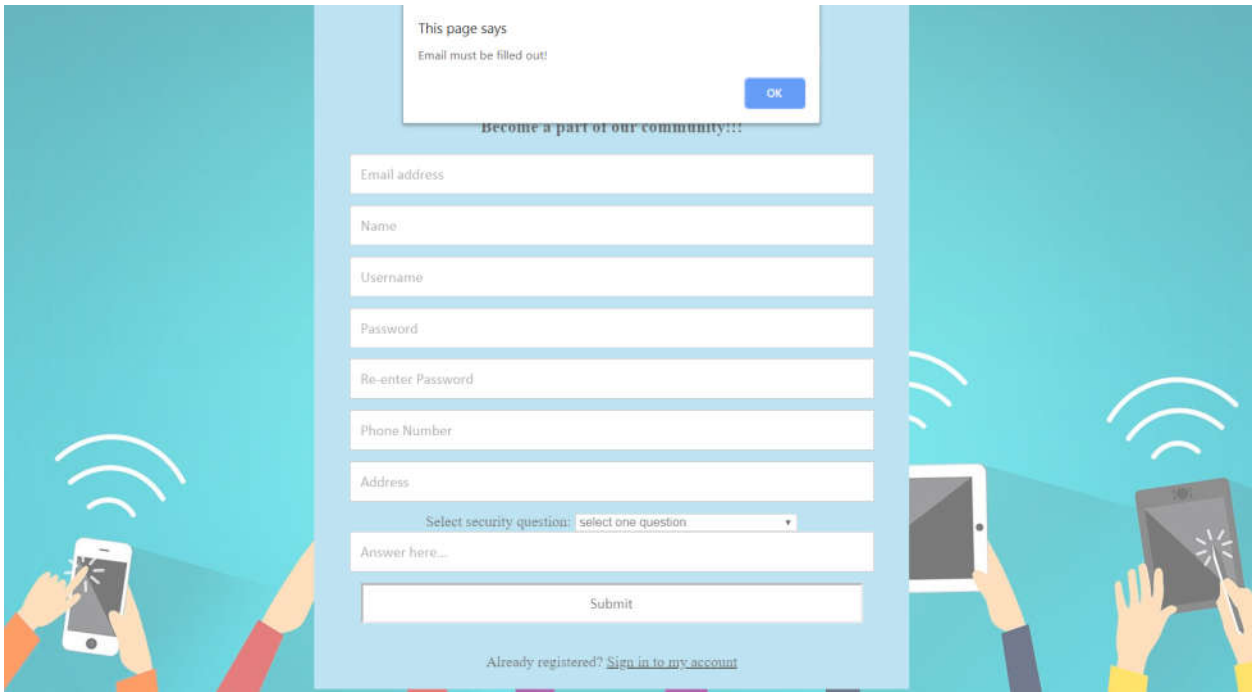

Contact Us
+0123456789
abc@abc.com
Cape Girardeau, Missouri

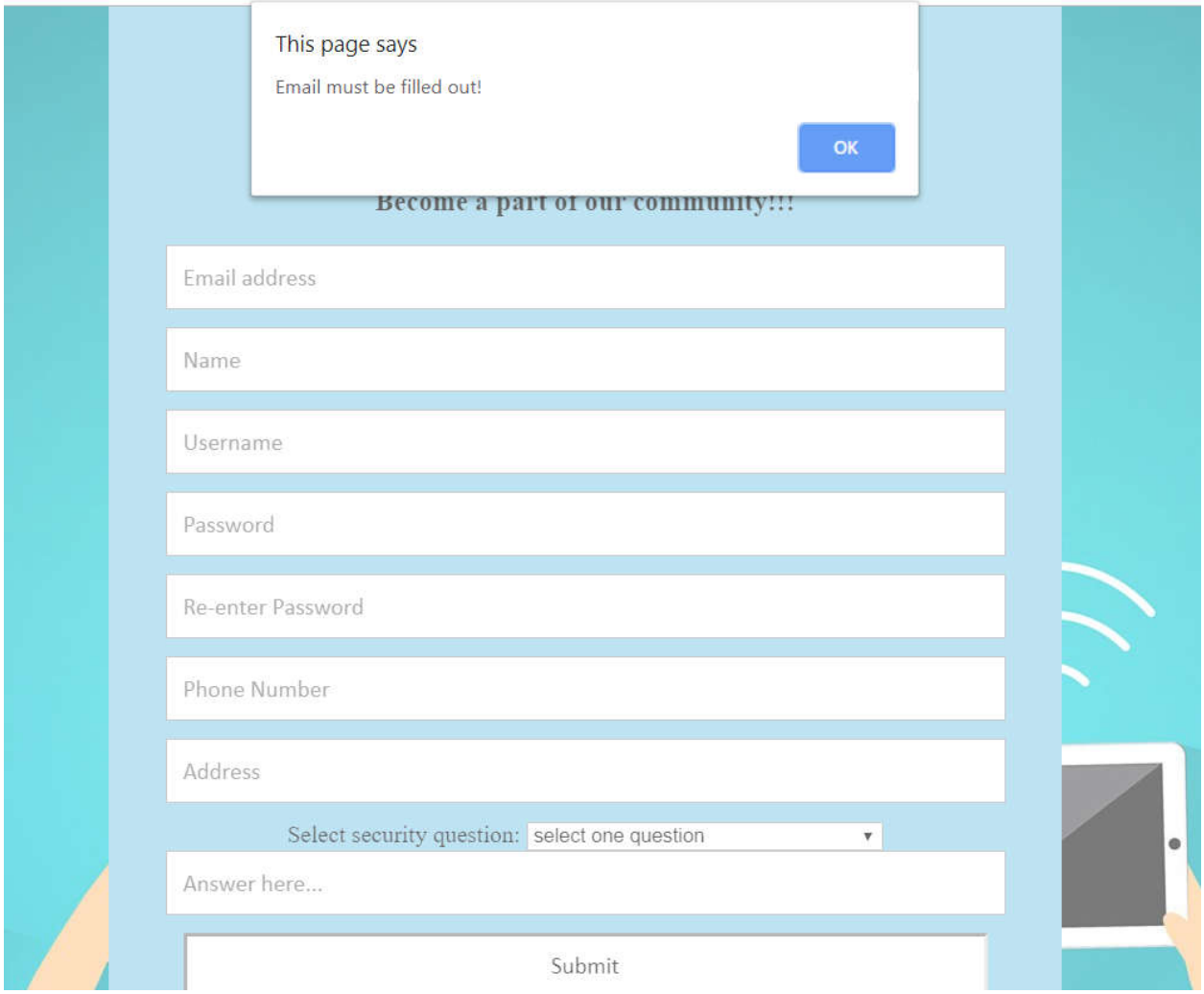

Smart Device Repair Handling System
Services
Feedback
Contact Us
Log In
Sign Up
Search
Search
Service Request
Track Request
Smart Home
Security
Lighting
Network
Camera
40%
21°C
69.8°F
09:37
ON
37°C
98.6°F


New Service Request
Name of the customer
Full Name
Select Age
Age
Enter contact No.
123-456-6789
Enter customer email
abc@domain.com
Validate Email
Upload File
Choose device
Select device
Additional Comments
Select service type
Home
Showroom
Online
Select appointment date and time
mm/dd/yyyy
--:-- --
Please accept the terms and conditions
Submit

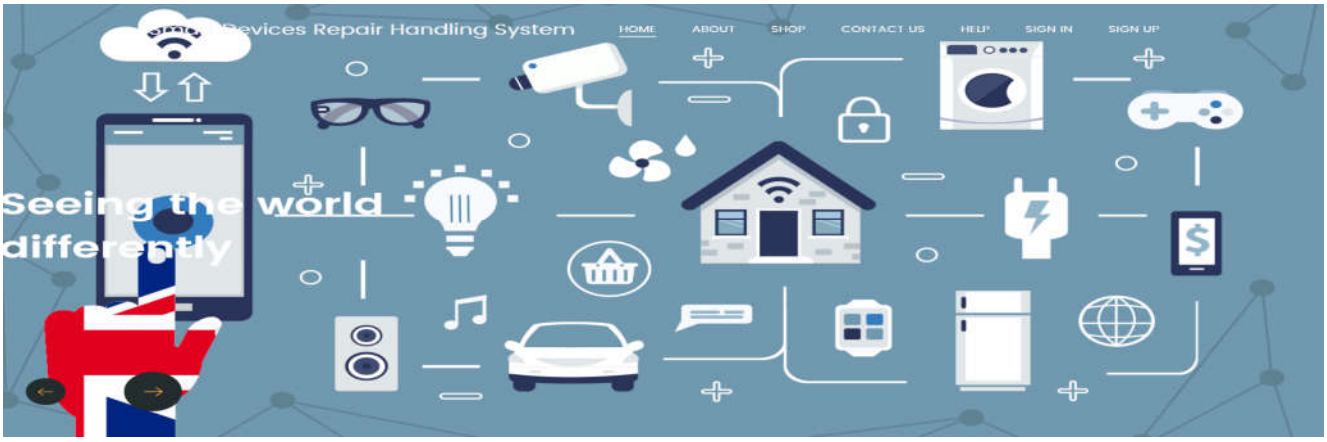

Tracking Information
Please enter your tracking Number:
Submit
View Tracking history: From date: 08/31/2019 To date: 08/31/2019
Submit to view History

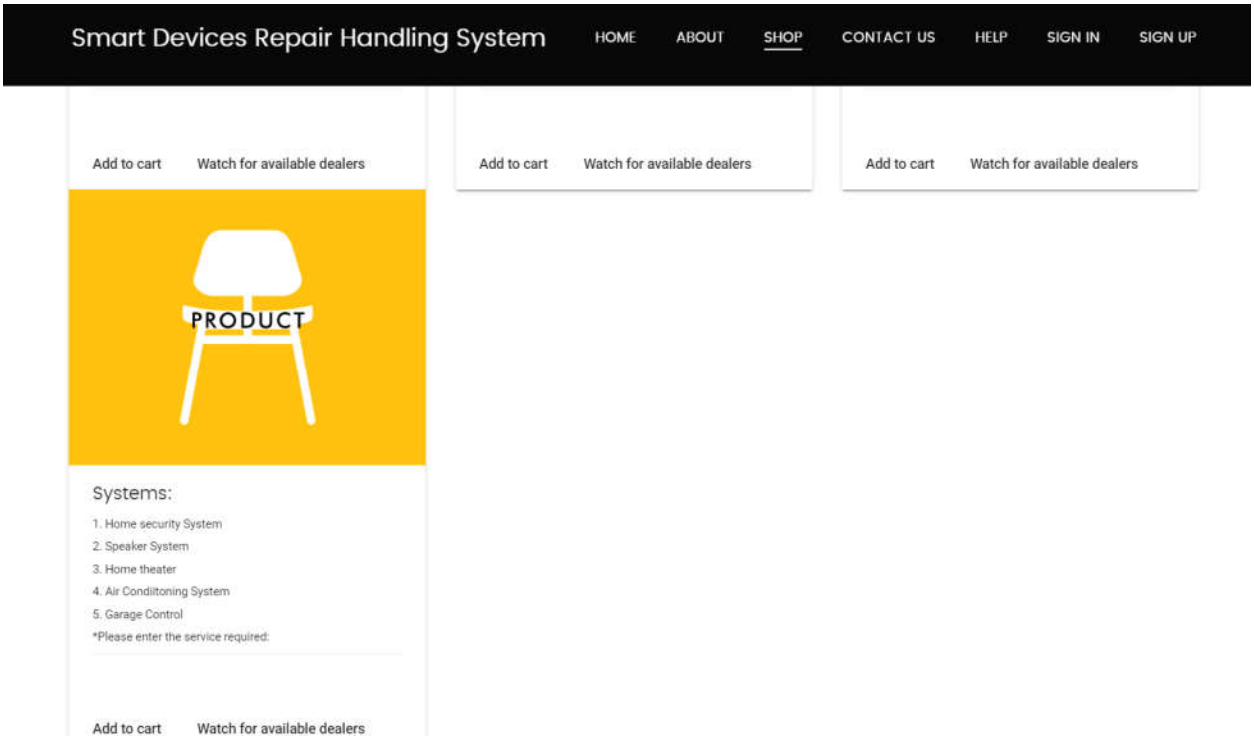

Feedback
Please provide your feedback below:
How do you rate your overall experience?
Extreamly Satisfied
Satisfied
Bad
Please enter your comments below
Your Comments
Your Name:
Email:
Submit Feedback

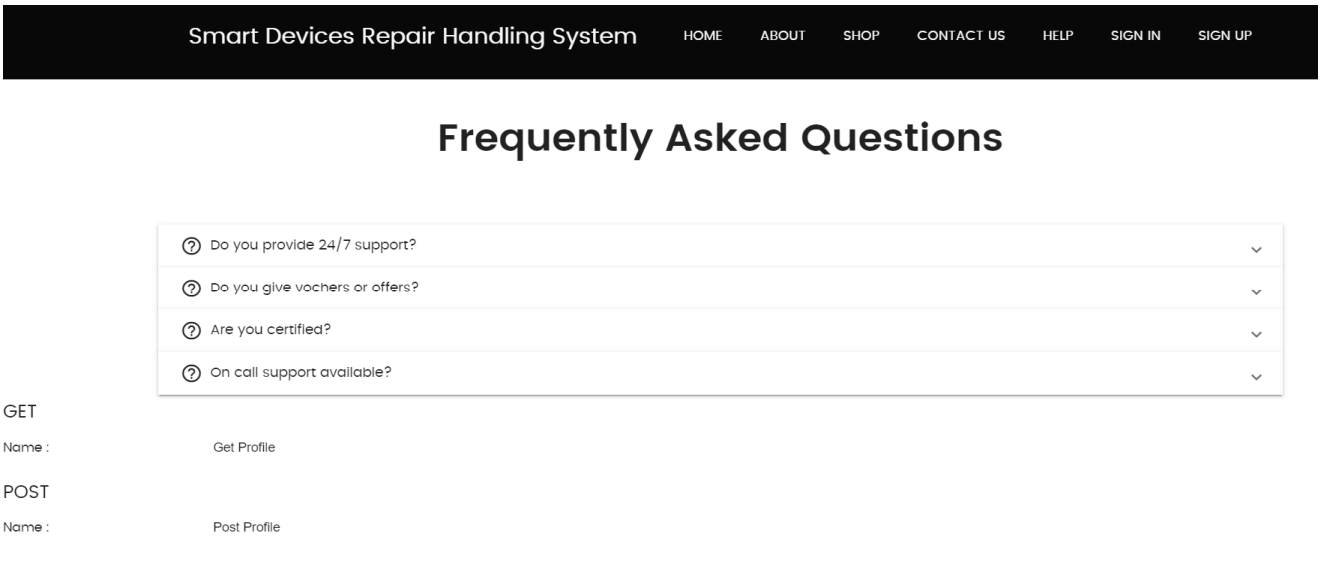

This page says
Email must be filled out!
OK
Become a part of our community!!!
Email address
Name
Username
Password
Re-enter Password
Phone Number
Address
Select security question:
select one question
Answer here...
Submit
Already registered? Sign in to my account

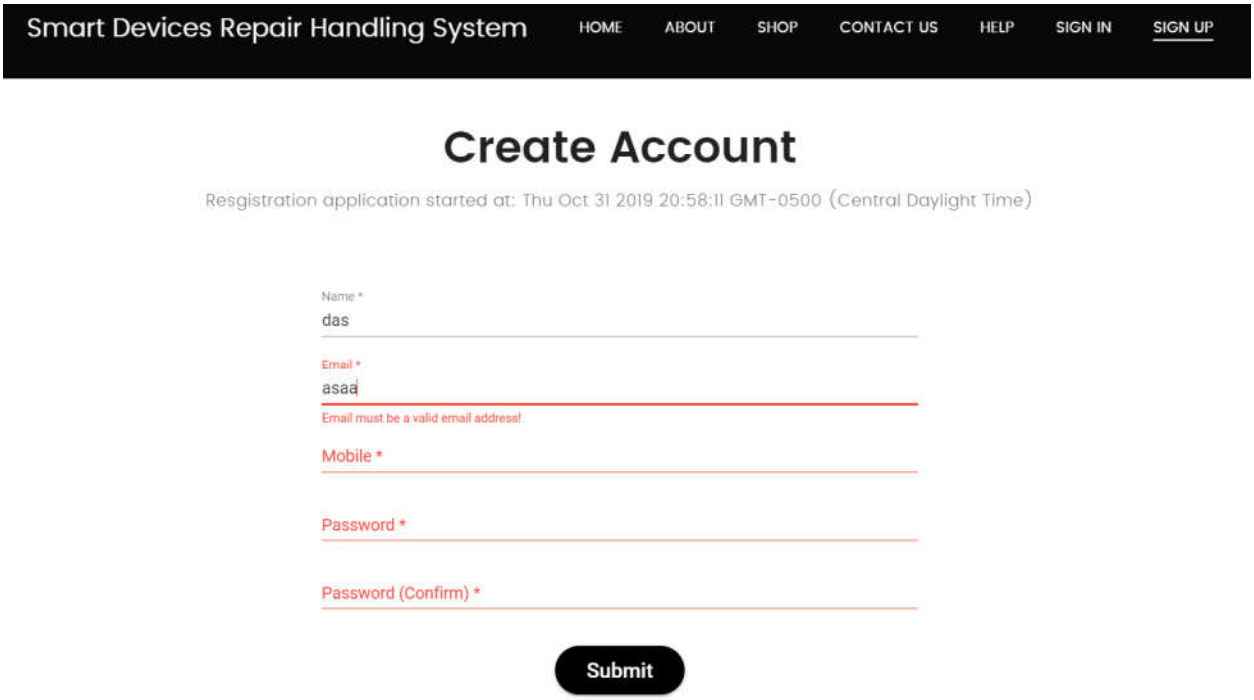


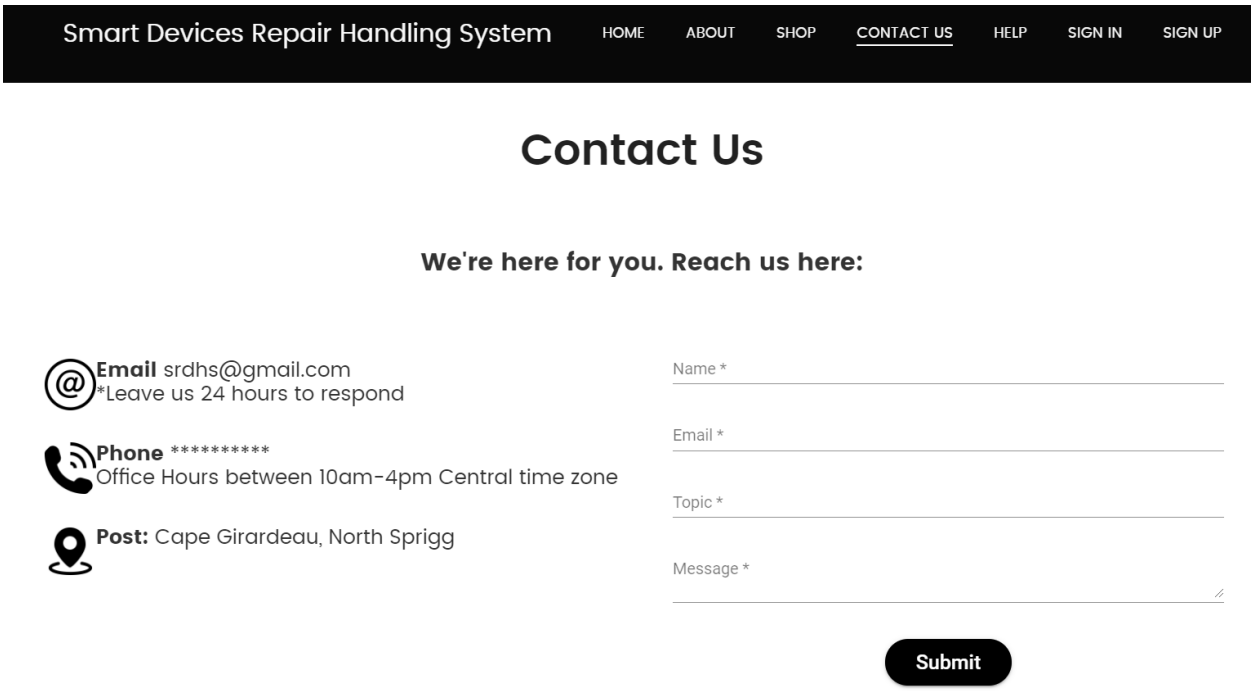
- **Angular Screenshots:**




Smart Devices Repair Handling System
HOME
ABOUT
SHOP
CONTACT US
HELP
SIGN IN
SIGN UP
Add to cart
Watch for available dealers
Add to cart
Watch for available dealers
Add to cart
Watch for available dealers
PRODUCT
Systems:
1. Home security System
2. Speaker System
3. Home theater
4. Air Conditoning System
5. Garage Control
*Please enter the service required:
Add to cart
Watch for available dealers


Smart Devices Repair Handling System
HOME
ABOUT
SHOP
CONTACT US
HELP
SIGN IN
SIGN UP
Frequently Asked Questions
Do you provide 24/7 support?
Do you give vochers or offers?
Are you certified?
On call support available?
GET
Name :
Get Profile
POST
Name :
Post Profile


Smart Devices Repair Handling System
HOME
ABOUT
SHOP
CONTACT US
HELP
SIGN IN
SIGN UP
Create Account
Resgistration application started at: Thu Oct 31 2019 20:58:11 GMT-0500 (Central Daylight Time)
Name *
das
Email *
asaa
Email must be a valid email address!
Mobile *
Password *
Password (Confirm) *
Submit


Smart Devices Repair Handling System
HOME
ABOUT
SHOP
CONTACT US
HELP
SIGN IN
SIGN UP
Contact Us
We're here for you. Reach us here:
Email srdhs@gmail.com
*Leave us 24 hours to respond
Phone **********
Office Hours between 10am-4pm Central time zone
Post: Cape Girardeau, North Sprigg
Name *
Email *
Topic *
Message *
Submit

Project 3: Evaluation of public and private cloud services for building and deploying an online reservation system in cloud environment

Name of the Professor: Dr. Ziping Liu

- Abstract: This project aims at providing the high-level system architecture along with the detailed elaboration of individual components involved in the system development. The application is implemented on public and private platforms using AWS (Amazon Web Services) and Open stack respectively. This report clearly depicts the costs incurred in procuring and establishing the system in both the environments and gives a comparison of both the system costs to evaluate and conclude on the cloud platform that perfectly adopts to the business objective of minimizing the cost for online reservation system.

- Goal: The main goal of the project is to evaluate the development of online reservation system in both public and private cloud systems thus minimizing the cost of infrastructure on the company.

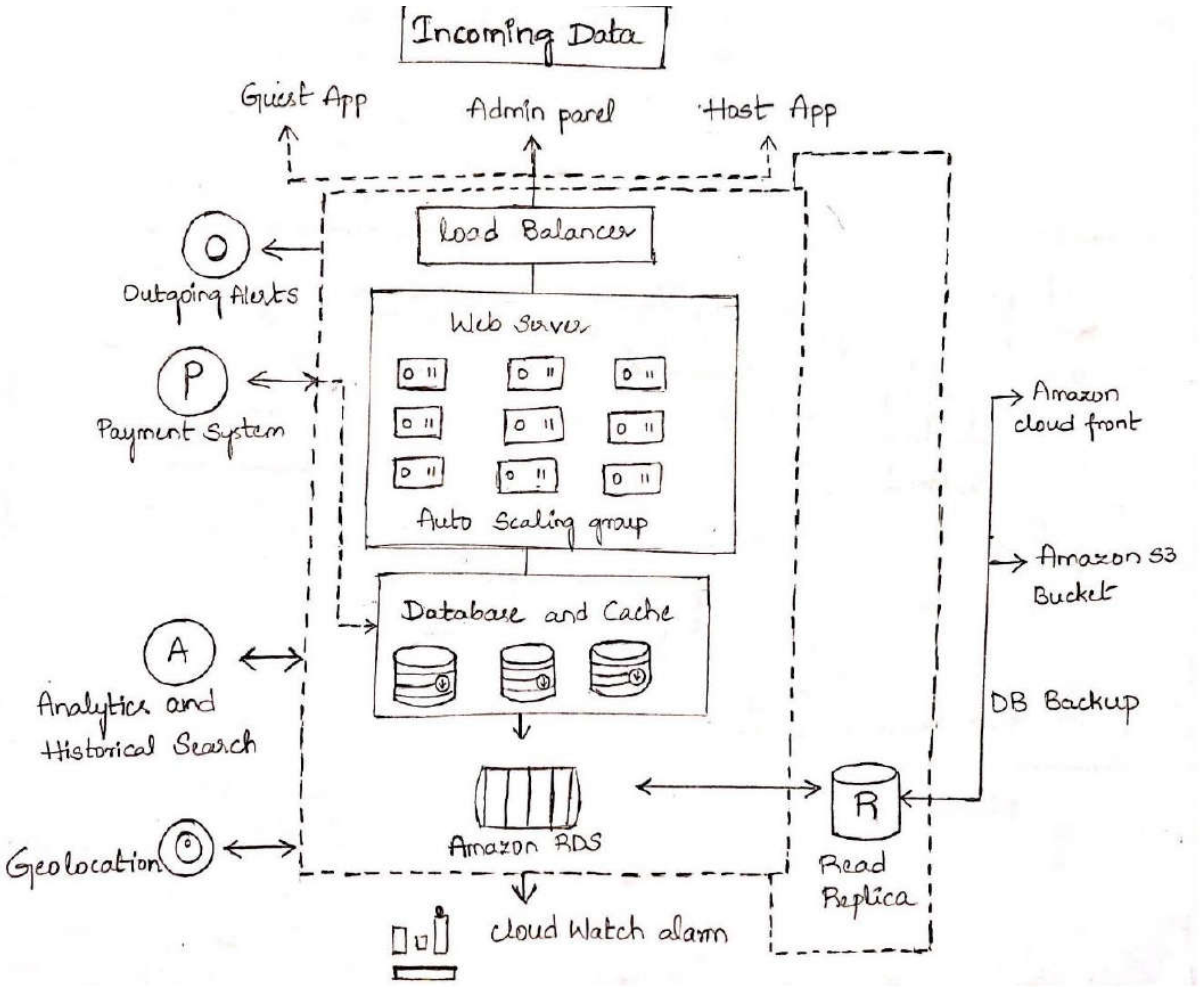
- Motivation: The motivation is to understand and compare private and public clouds while proposing the business model for online reservation system.
- Solution and Contribution: Orchestration in open stack and orchestration engine implementation helps in creating a human and machine accessible services for managing the complete project life cycle. Proposed High level architectural design, Identification of hardware and software requirements, Cloud services identification and framework design.
- Tools and Technologies: Amazon Web Services, RedHat OpenShift, RedHat OpenStack
- Deployment: These include Memcached, Repository that holds the reference set of artifacts like python wheels, load balancer, utility container, Log aggregation host, Unbound DNS container and Hypervisor configuration.

- Key results and Validation: When comparing the deployment platform for both private and public cloud, the active architecture of the web platform deployed the cloud design in the public cloud in an expensive way initially supporting a platform that includes deployment scenarios and use cases. The service demands high data flow from various API's and high-performance service orchestration for scalability which is expensive when deployed in the public cloud services like AWS. Also, users have an option to use a managed service offering as well buy hardware to run with an OpenStack cloud providing much more flexibility in terms of provisioning of services and infrastructure as per the usage and pay for only the services that are essential and required.

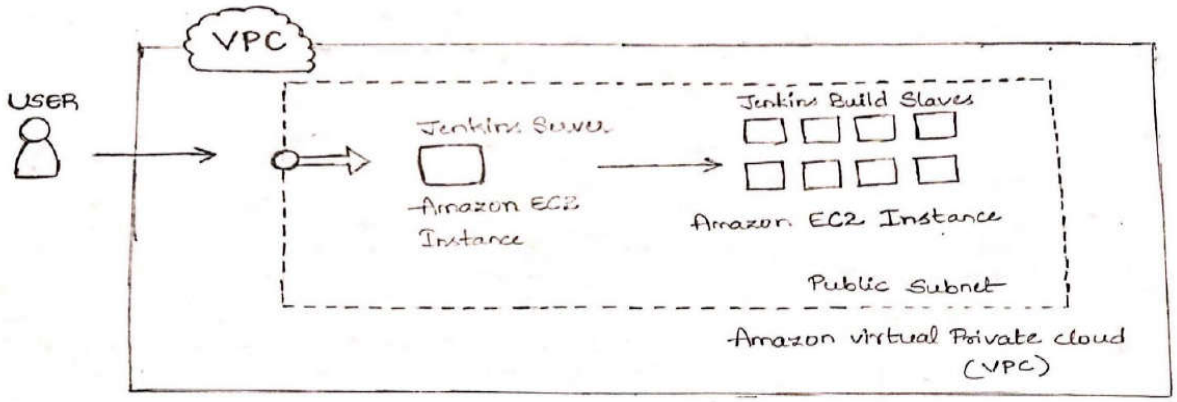
- **Keywords:** Public Cloud, Private Cloud, OpenStack, Amazon Web Services(AWS), DNS, Container, Orchestration, Load Balancer, Scalability, Application Programming Interface.

Block Diagrams:

- **General Architecture**

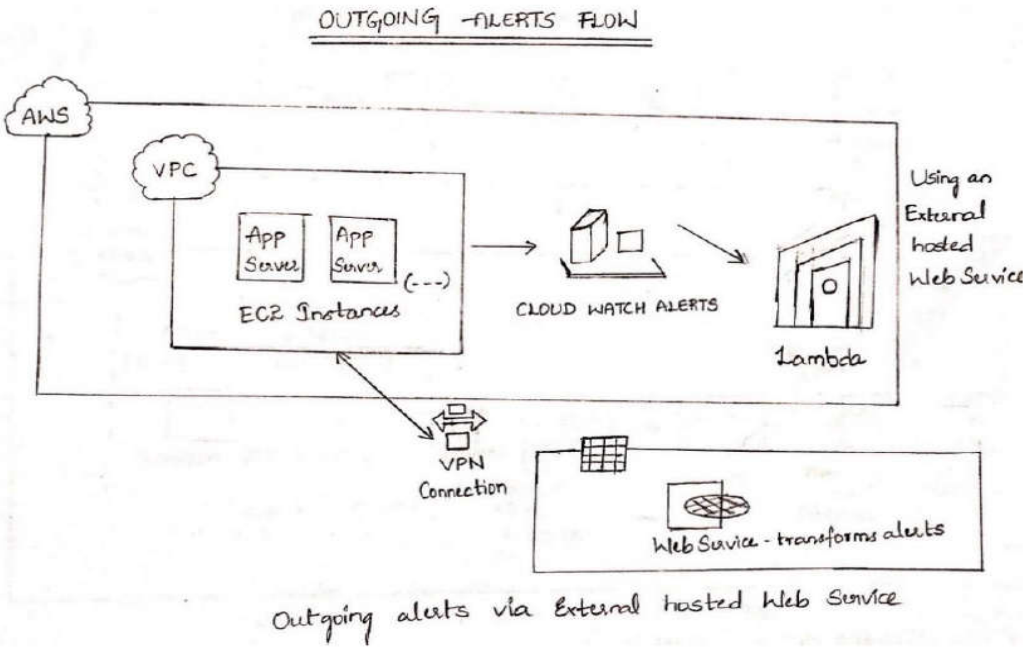


- **Server Setup and Auto Scaling group**



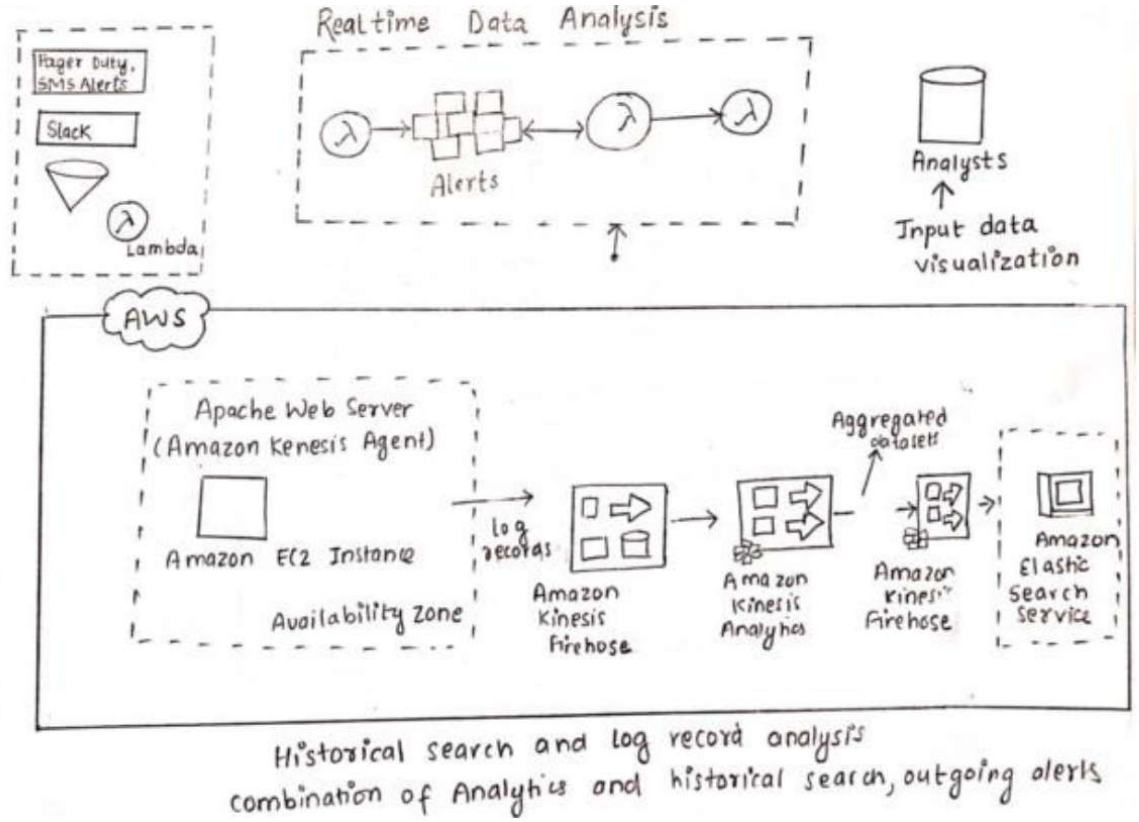
SERVER SETUP AUTO SCALING GROUP

- **Outgoing Alerts Flow**



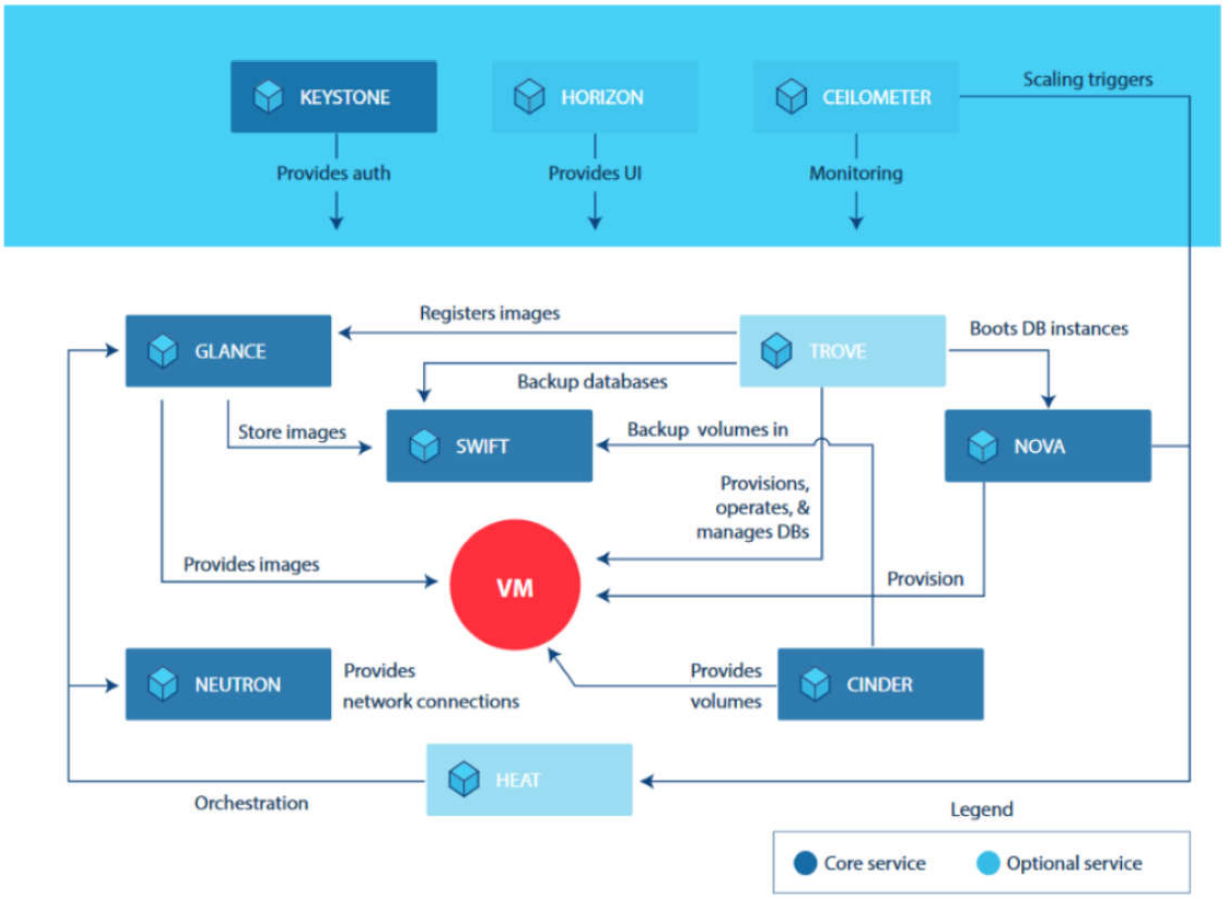
Outgoing alerts via External hosted Web Service

- **Real-time Data Analysis**



Historical search and log record analysis
combination of Analytics and historical search, outgoing alerts

- **OpenStack Architecture**



- **Public cloud deployment cost as a final estimate**

Cost of Deployment on a Public Cloud

		Public Cloud		
S. no.	Service Name	Cost for Instance	No. of Instance	Cummulative Cost
1	Lambda Service	0.000000208	4	0.000000832
2	Amazon EC2 Instance(t3.nano,Vcpu:2)	$0.0098 per Hour	4	$0.00392per Hour
3	Amazon Kinesis Firehose	0.029	3	0.087
4	Amazon Kinesis Analytics	$0.110 per hour	5	0.55
5	Amazon Elastic Search Service (t2.micro.elasticsearch)	$0.018 per Hour	4	$0.072 per Hour
6	Cloud Watch Alerts	$0.10 per alarm per month	3	0.3
7	Amazon S3	$0.02	4	$0.08
8	AWS Virtual Private Cloud(VPC)	$0.05 per VPN Connection-hour	4	0.2
9	AWS Simple Notification Service(SNS)	$0.50	2	$1
10	Cloud front	$0.09	3	0.027
12	Amazon EMR	$0.048	3	
13	Amazon Elastic search service	0.018	4	0.072

Research paper: Blockchain Implementation to Prevent Fraud Control in Banking Transactions

- Abstract: This paper gives the introduction to the blockchain technology and proceeds further on the Proof of Authorization and Proof of Work mechanisms. It then later explains the general banking system based on centralized databases. The paper then introduces the novice users about the blockchain formation, hash calculation, nonce formation, Merkle root and Tree Calculation. It then connects the concepts of consensus algorithm of blockchain for handling the fraud control in banking transactions thus preserving the confidentiality, integrity and availability of the databases and the data.

- Goal: To implement blockchain in banking system to have control on overcoming fraud transactions in banking.

- Motivation: Motivation is to understand blockchain, banking architecture, Proof of Work and Proof of Authentication, Hash Computation and Consensus Algorithm.

- Conclusions drawn: This technology can be used in fraud control. Since traditional banking systems have a centralized database which is vulnerable and more susceptible to cyber-attacks as all the data is located centrally. In the case of a blockchain system, there consists of a system which has features like distributed ledger, transparency, decentralized system that prevents fraud extensively. It is difficult to hack the transaction mechanism as many copies of the transaction's records are stored in different computers in a larger group of peers which makes difficult for the attacker to alter or completely remove the transaction. The block linking mechanism in blockchain helps to track the breach and can be prevented from being exploited.

  Also, fraudulent data cannot be inserted in the blockchain which prevents the risk of fake data and double transactions in the linked blockchain process and makes the accuracy of payments simpler. In addition to that, if blockchain is implemented in the banking system, any kind of transaction can be made real quick avoiding risks like data loss and problem in data synchronization which can increase the scalability of transactions. Thus, blockchain technology has the potential to improve financial transaction processes as well as has the potential to keep user data safe from different sources of fraud.

- **Keywords:** Block, Peer-to-Peer, Consensus Algorithm, Hash Function, Cryptocurrency, Fraud Control, SHA256, Centralized System, Cryptographic nonce, Distributed Computing, Distributed Ledger, C-IA Trait, Proof of Work, Proof of Authentication, Mining, Distributed Database, Distributed Ledger Technology (DLT).

Name of the Professor: Dr. Nick Rahimi

Algorithm:

- **Proof of work:** Proof-of-work is the original consensus algorithm in blockchain technology which is also known as mining process. In blockchain technology, all transactions are timestamped, and a distributed timestamp server needs to be implemented on a peer-to-peer network to confirm the transaction process and produce new blocks in the chain which is the Proof-of-work. In this technology, a transaction is verified and a complex mathematical puzzle that is associated with the block that is being created is verified by the miners.
- **Proof of Authentication:** A new consensus algorithm for blockchain Proof-of-Authentication (PoAh) helps to make blockchain suitable for distributed systems by authenticating the blocks. The proof-of-authentication explained here follows conventional blockchain model with block verification. It targets to authenticate the blocks that follow the same method of a transaction of blockchain.

Hash Algorithm:

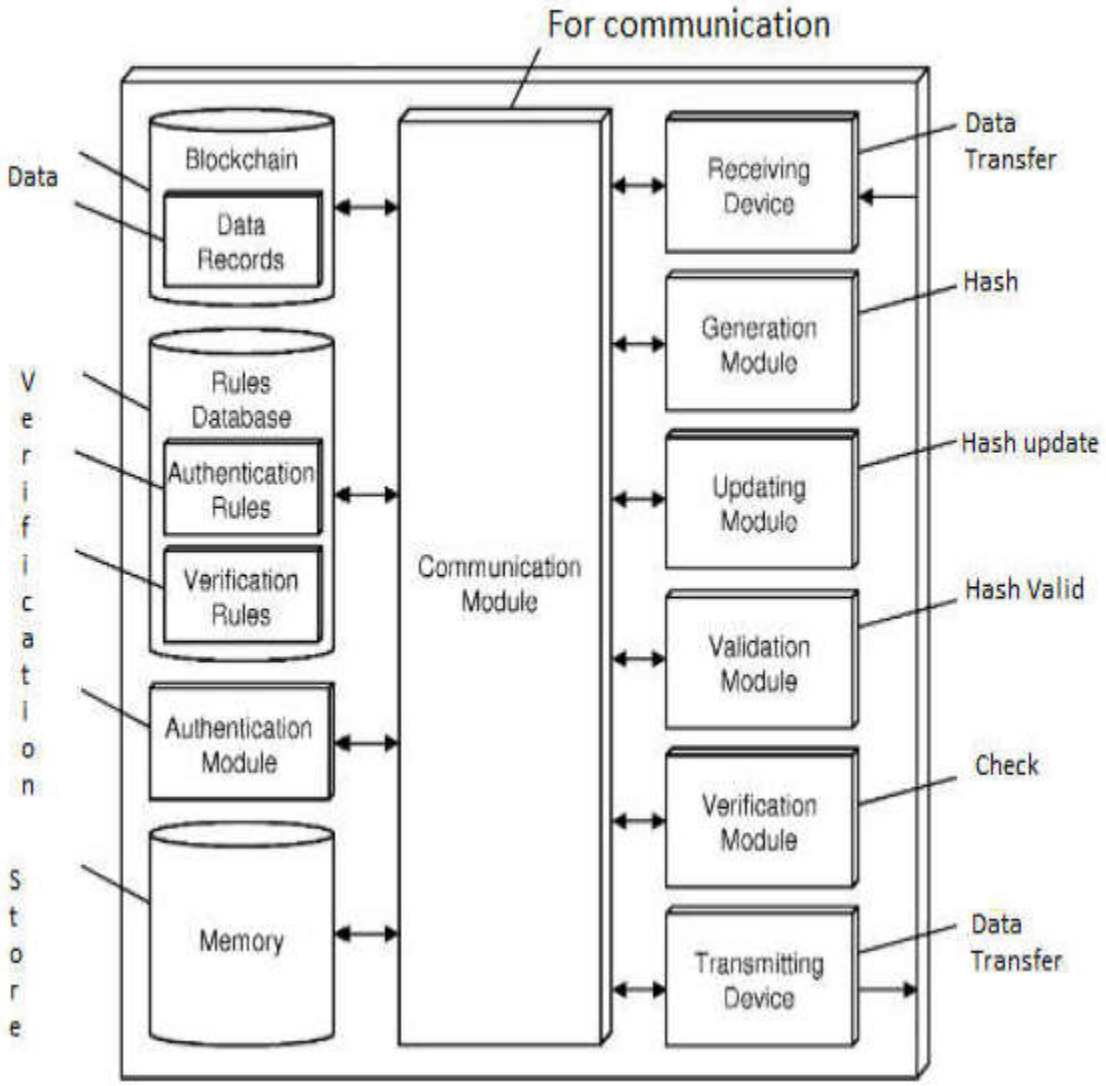
Algorithm

1. Start
2. Select the block and open
3. Compute the previous block hash
4. IF hash converges with the previously calculated hash block, THEN create a new block.
5. If not, End process
6. Once a new block is created, Transition it to consensus algorithm.
7. IF there is a conflict, Resolve the conflict and write a block in the Blockchain to end the process.
8. If there is no conflict Just write the block in Blockchain and end the process.

SHA Generation:

For byte_block in iter(lambda: f.read(4096),b""):
sha256_hash.update(byte_block)

Block Diagram:



Technical Report: Container Orchestration in Cloud Computing

- Abstract: This technical report introduces container orchestration mechanism and its architecture. The core components and their container functionalities are studied. The report then evaluates 10 orchestration systems available in the market based on varied parameters thus deducting valuable insights from each tabular form in order to make a recommendation on the best adoptive orchestration system. It then lists critical points that the authors missed while evaluating which when included would make the research on container orchestration systems more viable and adaptive across multiple applications.

- Goal: The goal of the report is to understand the container orchestration mechanism with focus on evaluating the different container orchestration systems that are more viable and are adaptive across multiple applications.
- Motivation: Since a platform as a service(PaaS) executes the workflows from the plan through a container engine like agents, The motivation was to understand the container and orchestration architecture while evaluating the container systems like Borg, Kubernetes, Swarm, Mesos, Aurora etc. across various parameters like Container Technology, Workload, Resources, Cluster Infrastructure, Elasticity, Architecture etc.

- Conclusions drawn: To conclude, Although Docker Swarm and Kubernetes provide best orchestration systems due to features like support for running all jobs and collocated tasks with efficient cluster elasticity and infrastructure, due to lack of resource estimation and rescheduling, Kubernetes loses to docker swarm in that it provides better scalability, availability, utilization and throughput comparatively.

- Critique: The study although enlightened on the methodology of container orchestration and could study available systems, the research can still be criticized in the way that the most adoptable Docker system mounts writable file on top of the readable base images.
  - The authors in either of the references have not deducted or stated any methodology to adopt, if we want to write data to the underlying read only files.
  - There has been no focus on node failure management, data storage limits that the orchestration systems possess, level of automation, cluster load distribution.
  - The authors have not considered security of applications w.r.t DoS attacks and its mitigation to the clusters. This needs critical attention in understanding orchestration.
  - Next important factors that needs much attention and has been ignored by the authors are resources cost, pricing model and the amount of man hours for tuning their virtual cluster.

- **Keywords:** Orchestration, Cloud, Containers, DoS(Denial of Service) attacks, Docker, Distributed systems.

Name of the Professor: Dr. Sumanth Yenduri

Classification Tables:

System	Container Technology	Workload	Job Composition	Cluster Elasticity	Cluster Infrastructure
Borg	Linux	All jobs	Independent	Static	Nonvirtualized
Kubernetes	Docker, API, OCI	All jobs	Collocated tasks	E,M,A	Virtualized, Nonvirtualized
Swarm	Docker	Long running	Collocated tasks	E,M	Virtualized, Nonvirtualized
Mesos	Mesos, Docker	All jobs	Single task	E,M	Virtualized, Nonvirtualized
Aurora	Mesos, Docker	Long, Cron	Independent	E,M	Virtualized, Nonvirtualized
Marathon	Mesos, Docker	Long	Collocated tasks	E,M	Virtualized, Nonvirtualized
YARN	Linux, Docker	Batch jobs	Single task	E,M	Virtualized, Nonvirtualized
Omega	--	All	Independent	Static	Nonvirtualized
Apollo	--	Batch jobs	Task graphs	Static	Nonvirtualized
Fuxi	Linux	Batch jobs	Task graphs	Static	Nonvirtualized

Table: 1

System	Quota Management	Resource Reclamation	Resource Granularity	Resource Estimation	Rescheduling	Architecture
Borg	Limit, Requests	Eviction, throttling	Fine-Grained	✓	✓	Centralized Monolithic
Kubernetes	Limit, Requests	Eviction, throttling	Fine-Grained	--	--	Decentralized Monolithic
Swarm	Requests	Eviction	Fine-Grained	--	✓	Decentralized Monolithic
Mesos	Requests	Eviction, throttling	Fine-Grained	--	--	Two-level offer-based
Aurora	Limits	Eviction, throttling	Fine-Grained	--	✓	Two-level offer-based
Marathon	Requests	Eviction, throttling	Fine-Grained	--	✓	Two-level offer-based
YARN	Requests	Eviction	Coarse-Grained	--	--	Two-level offer-based
Omega	--	--	Fine-Grained	--	✓	Decentralized Monolithic
Apollo	Limits	Eviction, throttling	Fine-Grained	--	✓	Decentralized Monolithic
Fuxi	Requests	Eviction	Bundle	--	✓	Two-level offer-based

Table 2

System	Scalability	Availability	Utilization	Throughput	Application QoS
Borg	✓	✓	✓	✓	--
Kubernetes	✓	--	--	–	--
Swarm	✓	✓	✓	✓	--
Mesos	✓	✓	--	–	--
Aurora	✓	✓	--	–	--
Marathon	✓	✓	--	–	--
YARN	✓	✓	--	✓	--
Omega	✓	✓	✓	✓	--
Apollo	✓	--	✓	✓	✓
Fuxi	✓	✓	✓	✓	--

Table 3

Future Work:

- The most adoptable Docker system mounts writable file on top of the readable base images. The authors in either of the references have not deducted or stated any methodology to adopt, if we want to write data to the underlying read only files.
- The focus of all the papers was on deployment of clusters but not on their development.
- The authors have paid no attention to the challenges with node failure management and advanced networking support which play a major role in distribution systems.
- There has been no focus on data storage limits that the orchestration systems possess.
- The level of automation, cluster load distribution that these systems leverage is not addressed by the authors.
- The authors have not considered security of applications w.r.t DoS attacks and its mitigation to the clusters. This needs critical attention in understanding orchestration[5].
- Next important factors that needs much attention and has been ignored by the authors are resources cost, pricing model and the amount of man hours for tuning their virtual cluster.

The analysis of orchestration systems in the research papers would be complete if their evaluation with respect to the above factors were addressed.