Frankfurt University of Applied Sciences

Master: Information Technology

Course: Agile Development in Cloud Computing Environments - SoSe 22

Prof. Patrick Wacht

Topic: Travel Application Website (AWS Platform)

Submitted by:

Gaurav Honnavara Manjunath (1384178)

Manoj Kuridoddi Marigowda(1348114)

Date of submission: **June 29, 2022**

Table of Contents

1. Introduction

Agile software development has had a significant influence on how software development is conducted. It allows software developers to plan and coordinate their work, communicate with customers and external stakeholders, and organise software development in small, medium-sized, and large companies from the telecom and healthcare sectors to games and interactive media.

Some of the Agile approaches are

- ✓ Collaboration between the development team and business stakeholders
- ✓ Frequent delivery of business value
- ✓ Self-organizing teams
- ✓ Innovative ways to create, test and deploy code

2. Kanban

The Kanban method approaches incremental, evolutionary processes and systems change for organizations.

Kanban can be described as

- ✓ A way to organize the chaos surrounding so many delivery teams is by clarifying the need for prioritization and focus.
- ✓ A way to uncover workflow and process problems so you may solve them to deliver more consistently to your client/customer/etc.

2.1 Principles Of Kanban

- ✓ Start with what you do now
- ✓ Agree to pursue incremental, evolutionary change
- ✓ Respect the current process, roles, responsibilities & titles
- ✓ Encourage acts of leadership at all levels

2.2 Core Properties of Kanban

- ✓ Visualize the workflow
- ✓ Limit WIP
- ✓ Manage flow
- ✓ Make Process Policies Explicit
- ✓ Improve Collaboratively (using models & the scientific method)

3. Scrum

Scrum is a framework for developing and sustaining complex products.
Scrum requires a Scrum Master to foster an environment where:

- ✓ A Product Owner orders the work for a complex problem into a Product Backlog.
- ✓ The Scrum Team turns a selection of the work into an Increment of value during a Sprint.
- ✓ The Scrum Team and its stakeholders inspect the results and adjust for the next Sprint.
- ✓ Repeat

4. User stories and Backlog

With customer Interaction, the team divides the work into small pieces called 'user stories'. Each story should be independent and is expected to produce a contribution of value to the overall product.

A backlog is a list of features or technical tasks the team maintains. The backlog is the primary entry point for knowledge about requirements and the single authoritative source defining the work to be done. The backlog is expected to change throughout the project's duration as the team gains knowledge.

5. Implementation

The project aims to develop and implement travel applications and hosting in Cloud (Amazon Web Services (AWS)). The project required the team to use agile methodology for development, including cross-platform integration such as Train Service, Flight Service, Taxi Service, Payment services and a Smart Robo adviser to advise the user using REST APIs. The Application also includes a user interface where our customers can register with the Application by signing up and logging in with stored data. The Application allows users to create a new request enabling the users to consume the RESTful services. The Project timeline was Planned for 4 Sprints, with each sprint interval of 2 weeks. Induvial Sprints involved tasks in Understanding Customer requirements, Development, Integrate, Testing and Delivering the product after the end of the project timeline.

Understanding the principles of Agile Methodology, the team planned different sprints to deliver the product. The project planning was done using Kanban Board.

5.1 Project Estimation

The Team initially worked on Understanding Customer Requirement, Project Requirement and follow the timeline to deliver the product within the speculated deadline.

The timeline was planned for 4 Sprints with each sprint interval of 2 weeks
With 2 resources in the team.

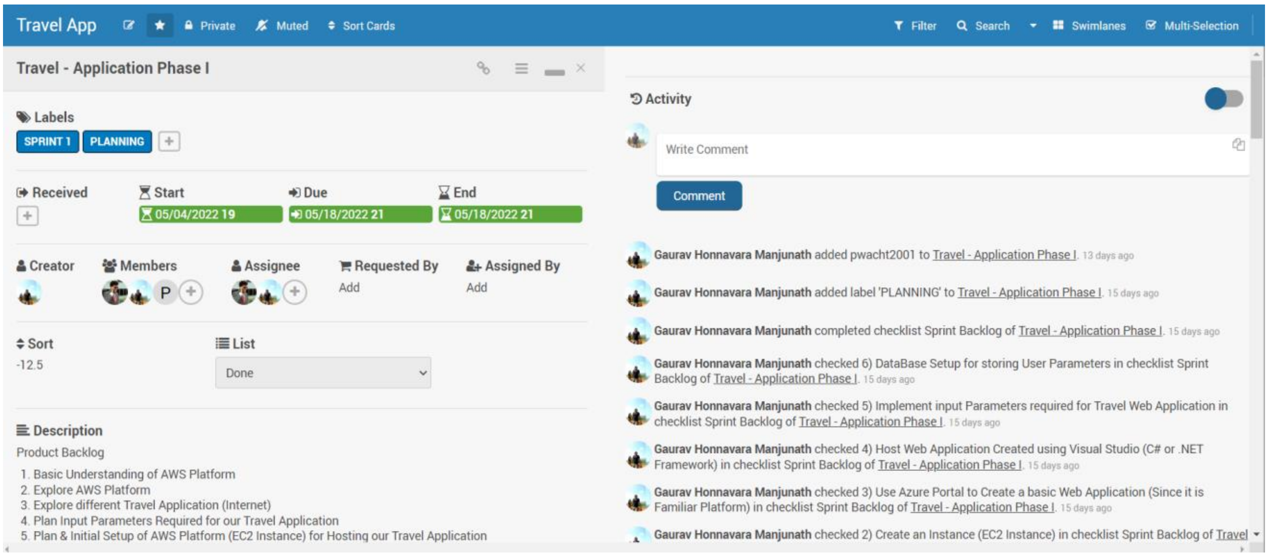
Project Estimation (Kanban Board Link): http://54.93.249.238/b/yzqpdYrwdaHfPRkPM/travel-app

5.1.1 Sprint 1 (Phase I)

Planning Phase (4.5.2022 - 18.5.2022)	
Product Backlog	Basic Understanding of AWS Platform
	Explore AWS Platform
	Explore different Travel Application (Internet)
	Plan Input Parameters Required for our Travel Application
	Plan & Initial Setup of AWS Platform (EC2 Instance) for Hosting our Travel Application
	Explore different servers (Apache, XAMAPP) & SQL database Required for our Website
	Initial Outlay of Travel Application using (HTML & CSS, or C# .Net FrameWork) Discuss with Teams to understand product requirement & Understand their Specification
Sprint Backlog	Kickstart -> Create AWS Account (Student Account)
	Create an Instance (EC2 Instance)
	Use Azure Portal to Create a basic Web Application (Since it is Familiar Platform)
	Host Web Application Created using Visual Studio (C# or .NET Framework)
	Implement input Parameters required for Travel Web Application
	Database Setup for storing User Parameters
Sprint Review	Created AWS Account and understand AWS Platform
	Able to capture Input Parameters Required for Our Travel Application
	Created EC2 instance in AWS and host a demo website using EC2 Instance
	Create a web Application using C# & .NET Framework and host using Azure
	Database Setup for Storing Input Parameters
	Discussed with Other Teams to Understand Product Requirement & Specification

Retrospective	Team was able to develop a basic web application using C# .NET Framework, but could foresee future risks and Incompatibility Issues if Product is developed using C# & .NET Framework (Discussed with Professor And Teammates and Come up with backup Plans)
	Team was able to host a basic web application in AWS Platform using static, eventually the team worked around to make the web application host dynamically

Table 1. Sprint 1 details – Kanban Board



Travel - Application Phase I

Labels
SPRINT 1 PLANNING

Received
Start 05/04/2022 19
Due 05/18/2022 21
End 05/18/2022 21

Creator
Members
Assignee
Requested By Add
Assigned By Add

Sort -12.5
List Done

Description
Product Backlog

1. Basic Understanding of AWS Platform
2. Explore AWS Platform
3. Explore different Travel Application (Internet)
4. Plan Input Parameters Required for our Travel Application
5. Plan & Initial Setup of AWS Platform (EC2 Instance) for Hosting our Travel Application

Activity

Gaurav Honnavara Manjunath added pwacht2001 to Travel - Application Phase I. 13 days ago

Gaurav Honnavara Manjunath added label 'PLANNING' to Travel - Application Phase I. 15 days ago

Gaurav Honnavara Manjunath completed checklist Sprint Backlog of Travel - Application Phase I. 15 days ago

Gaurav Honnavara Manjunath checked 6) DataBase Setup for storing User Parameters in checklist Sprint Backlog of Travel - Application Phase I. 15 days ago

Gaurav Honnavara Manjunath checked 5) Implement input Parameters required for Travel Web Application in checklist Sprint Backlog of Travel - Application Phase I. 15 days ago

Gaurav Honnavara Manjunath checked 4) Host Web Application Created using Visual Studio (C# or .NET Framework) in checklist Sprint Backlog of Travel - Application Phase I. 15 days ago

Gaurav Honnavara Manjunath checked 3) Use Azure Portal to Create a basic Web Application (Since it is Familiar Platform) in checklist Sprint Backlog of Travel - Application Phase I. 15 days ago

Gaurav Honnavara Manjunath checked 2) Create an Instance (EC2 Instance) in checklist Sprint Backlog of Travel

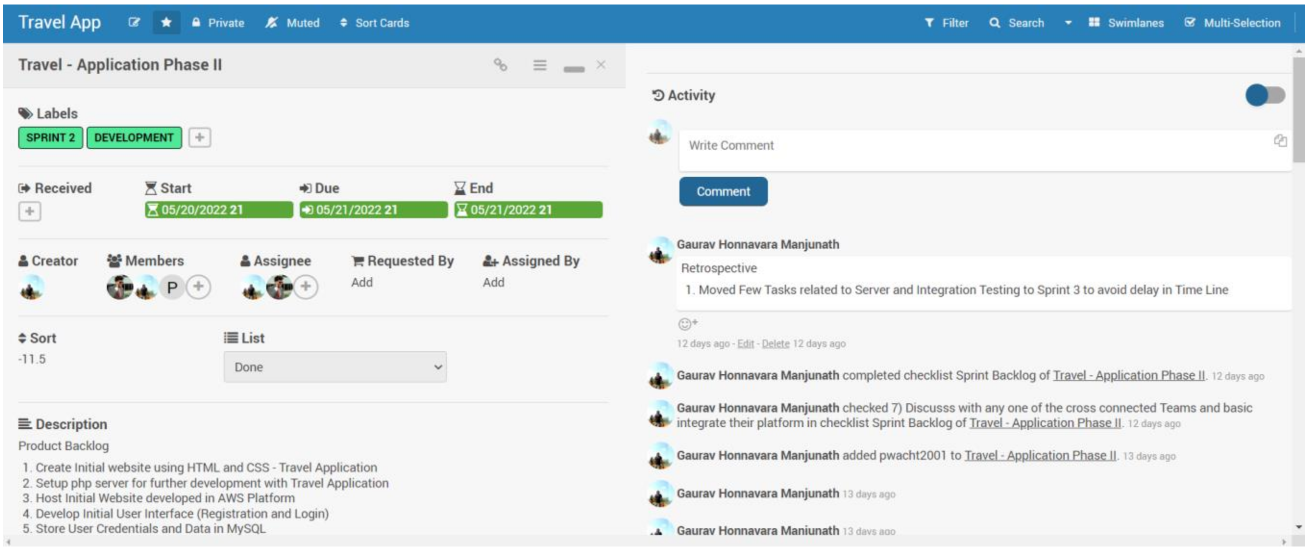
Fig 1. Sprint 1 details – Kanban Board

5.1.2 Sprint 2 (Phase II)

Development Phase (18.5.2022 - 1.6.2022)	
Product Backlog	Create Initial website using HTML and CSS - Travel Application
	Setup php server for further development with Travel Application
	Host Initial Website developed in AWS Platform
	Develop Initial User Interface (Registration and Login)
	Store User Credentials and Data in MySQL
	Discussion with Teams to be in line with product Integration & Specification Sprint Backlog
Sprint Backlog	Develop Initial Website using HTML& CSS using VS Code
	PHP Server Creation using Apache Server
	Hosting Initial Website in AWS Platform (Initial Website interface)
	Develop Registration Form, Login Page & UserInterface

	Test registration form, Login Page & User Interface
	Store User Credential and User Data in Database
	Discuss with any one of the cross connected Teams and basic integrate their platform
Sprint Review	Developed Initial Website using HTML & CSS
	Host Initial website in AWS Platform
	Developed Registration form, Login Page & User Interface
	Test Registration form, Login Page & User Interface
	Stored User Credential & User Data in Database
Retrospective	Delay in Interaction with Team for understanding Basic integration and Parameter
	Delay in start of Sprint 2 Development as Team was working around with a Plan which foresaw future risk and compatibility Issue
	Moved Few Tasks related to Server and Integration Testing to Sprint 3 to avoid delay in Timeline

Table 2. Sprint 2 details – Kanban Board



Travel - Application Phase II

Labels: SPRINT 2 DEVELOPMENT

Received: +

Start: 05/20/2022 21

Due: 05/21/2022 21

End: 05/21/2022 21

Creator

Members: P

Assignee

Requested By: Add

Assigned By: Add

Sort: -11.5

List: Done

Description

Product Backlog

1. Create Initial website using HTML and CSS - Travel Application
2. Setup php server for further development with Travel Application
3. Host Initial Website developed in AWS Platform
4. Develop Initial User Interface (Registration and Login)
5. Store User Credentials and Data in MySQL

Activity

Gaurav Honnavara Manjunath

Retrospective

1. Moved Few Tasks related to Server and Integration Testing to Sprint 3 to avoid delay in Time Line

12 days ago - Edit - Delete 12 days ago

Gaurav Honnavara Manjunath completed checklist Sprint Backlog of Travel - Application Phase II. 12 days ago

Gaurav Honnavara Manjunath checked 7) Discusss with any one of the cross connected Teams and basic integrate their platform in checklist Sprint Backlog of Travel - Application Phase II. 12 days ago

Gaurav Honnavara Manjunath added pwacht2001 to Travel - Application Phase II. 13 days ago

Gaurav Honnavara Manjunath 13 days ago

Gaurav Honnavara Manjunath 13 days ago

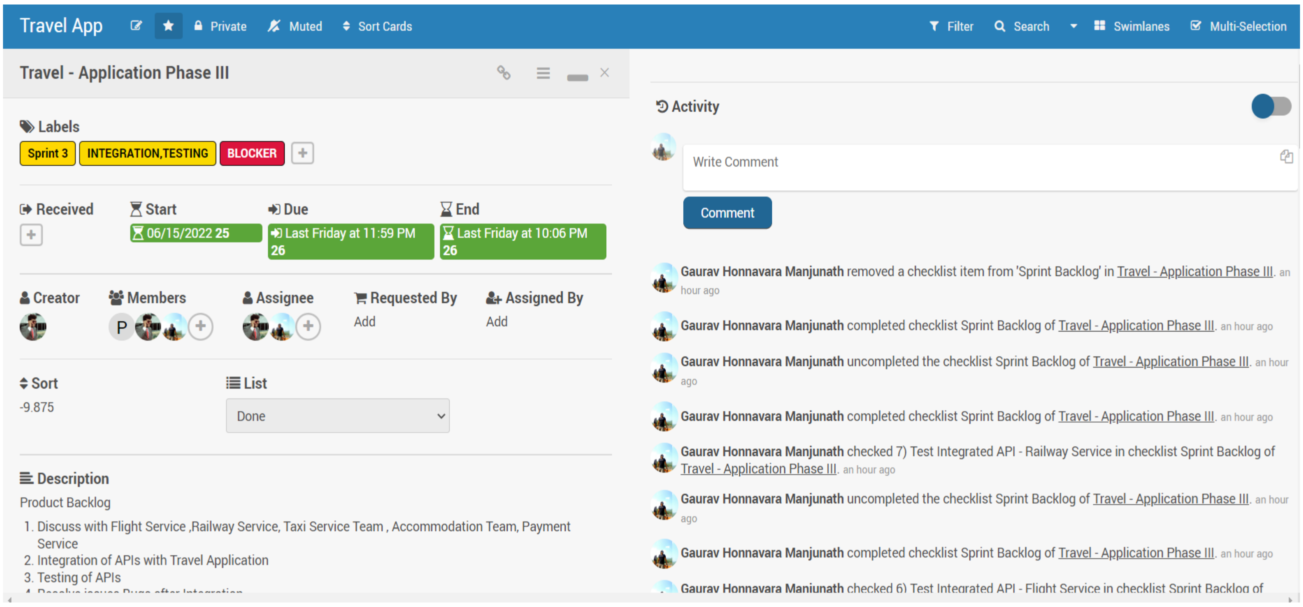
Fig 2. Sprint 2 details – Kanban Board

5.1.3 Sprint 3 (Phase III)

Integration & Testing Phase (2. 6.2022 - 15.6.2022)	
Product Backlog	Discuss with Railway Service & Flight Service Team
	Integration of APIs with Travel Application
	Testing of APIs
	Resolve issues Bugs after Integration
	Hosting Website in AWS

Sprint Backlog	Discussion with Flight Service & Railway Service Team
	Understand RESTful APIs Developed by Flight Service Team
	Understand RESTful APIs Developed by Railway Service Team
	Integrate RESTful API Developed by Flight Service Team
	Integrate RESTful APIs Developed by Railway Service Team
	Test Integrated API - Flight Service
	Test Integrated API - Railway Service
Sprint Review	Discussion with Flight Service & Railway Service Team
	Understand RESTful APIs Developed by Flight Service Team
	Understand RESTful APIs Developed by Railway Service Team
	Integration RESTful APIs Developed by Flight Service Team
	Testing RESTful APIs Developed by Flight Service Team
	Integration of RESTful APIs Developed by Railway Service Team
Retrospective	Opened Bug Sprint to take actions on Missing Items

Table 3. Sprint 3 details – Kanban Board



Travel - Application Phase III

Start: 06/15/2022 25

Due: Last Friday at 11:59 PM 26

End: Last Friday at 10:06 PM 26

Sort: -9.875

List: Done

Product Backlog

1. Discuss with Flight Service ,Railway Service, Taxi Service Team , Accommodation Team, Payment Service
2. Integration of APIs with Travel Application
3. Testing of APIs

Gaurav Honnavara Manjunath removed a checklist item from 'Sprint Backlog' in Travel - Application Phase III. an hour ago

Gaurav Honnavara Manjunath completed checklist Sprint Backlog of Travel - Application Phase III. an hour ago

Gaurav Honnavara Manjunath uncompleted the checklist Sprint Backlog of Travel - Application Phase III. an hour ago

Gaurav Honnavara Manjunath completed checklist Sprint Backlog of Travel - Application Phase III. an hour ago

Gaurav Honnavara Manjunath checked 7) Test Integrated API - Railway Service in checklist Sprint Backlog of Travel - Application Phase III. an hour ago

Gaurav Honnavara Manjunath uncompleted the checklist Sprint Backlog of Travel - Application Phase III. an hour ago

Gaurav Honnavara Manjunath completed checklist Sprint Backlog of Travel - Application Phase III. an hour ago

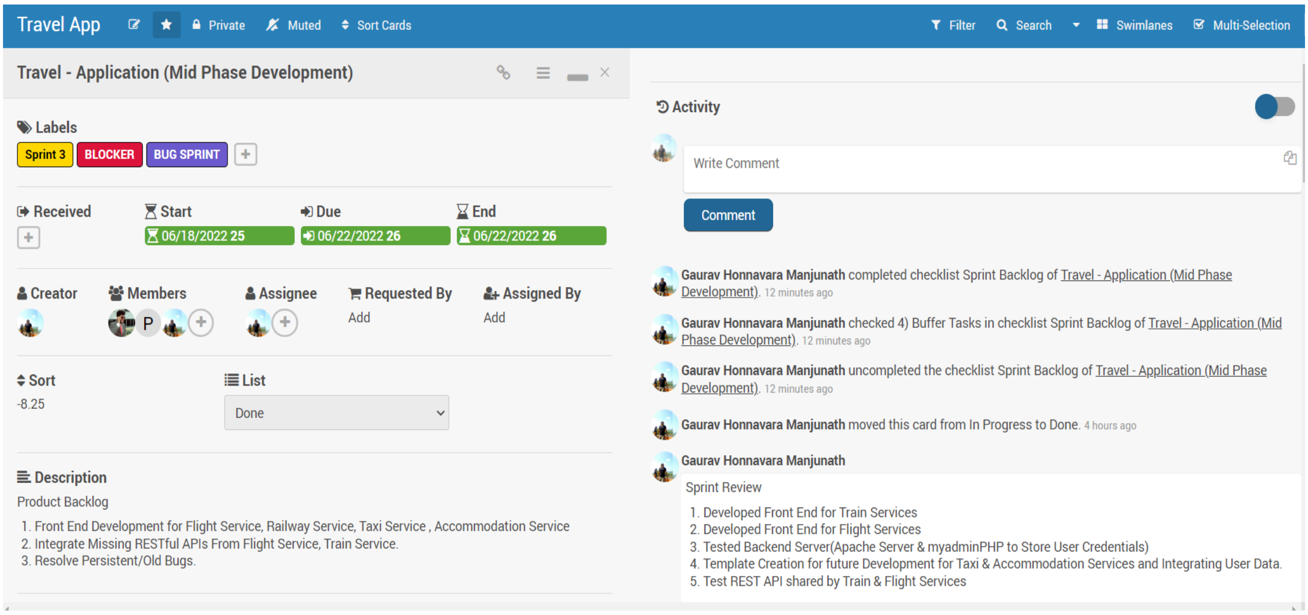
Gaurav Honnavara Manjunath checked 6) Test Integrated API - Flight Service in checklist Sprint Backlog of

Fig 3. Sprint 3 details – Kanban Board

Mid Phase Development (18. 6.2022 - 22.6.2022)	
Product Backlog	Front End Development for Flight Service, Railway Service, Taxi Service , Accommodation Service
	Integrate Missing RESTful APIs From Flight Service, Train Service.
	Resolve Persistent/Old Bugs.

Sprint Backlog	Front End Development for Flight Service
	Front End Development for Railway Service
	Resolve Bugs
Sprint Review	Developed Front End for Train Services
	Developed Front End for Flight Services
	Tested Backend Server (Apache Server & myadminPHP to Store User Credentials)
	Template Creation for future Development for Taxi & Accommodation Services and Integrating User Data.
	Test REST API shared by Train & Flight Services
Retrospective	Delay in Delivering APIs from Cross platform from Team Accommodation Services.
	Less Time Constraint led to Integration of Functional Aspects of Flight Service, Train Service and User Credentials.
	APIs from Taxi Service and Payment Service was not Delivered

Table 4. Sprint 3(Mid-Phase Sprint) details – Kanban Board



Travel - Application (Mid Phase Development)

Labels: Sprint 3 BLOCKER BUG SPRINT

Received

Start: 06/18/2022 25

Due: 06/22/2022 26

End: 06/22/2022 26

Creator

Members P

Assignee

Requested By: Add

Assigned By: Add

Sort: -8.25

List: Done

Description

Product Backlog

1. Front End Development for Flight Service, Railway Service, Taxi Service , Accommodation Service
2. Integrate Missing RESTful APIs From Flight Service, Train Service.
3. Resolve Persistent/Old Bugs.

Gaurav Honnavara Manjunath completed checklist Sprint Backlog of Travel - Application (Mid Phase Development). 12 minutes ago

Gaurav Honnavara Manjunath checked 4) Buffer Tasks in checklist Sprint Backlog of Travel - Application (Mid Phase Development). 12 minutes ago

Gaurav Honnavara Manjunath uncompleted the checklist Sprint Backlog of Travel - Application (Mid Phase Development). 12 minutes ago

Gaurav Honnavara Manjunath moved this card from In Progress to Done. 4 hours ago

Gaurav Honnavara Manjunath

Sprint Review

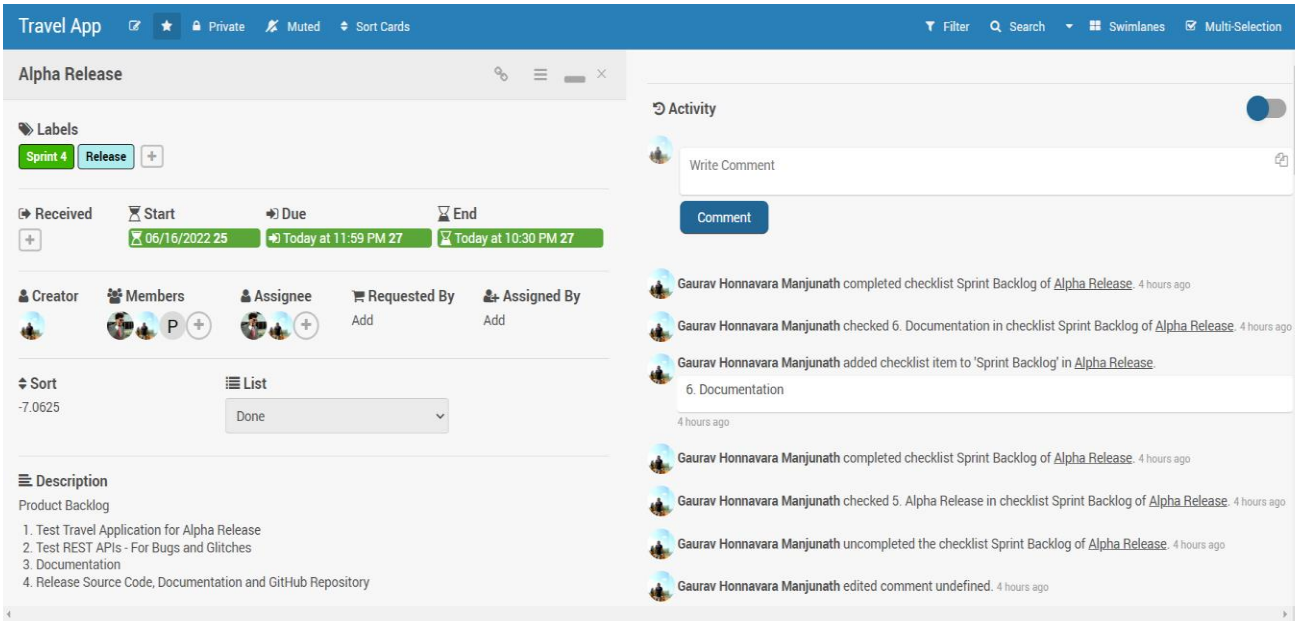
1. Developed Front End for Train Services
2. Developed Front End for Flight Services
3. Tested Backend Server(Apache Server & myadminPHP to Store User Credentials)
4. Template Creation for future Development for Taxi & Accommodation Services and Integrating User Data.
5. Test REST API shared by Train & Flight Services

Fig 4. Sprint 3(Mid-Phase Sprint) details – Kanban Board

5.1.4 Sprint 4 (Phase IV)

Alpha Release (16. 6.2022 - 29.6.2022)	
Product Backlog	Test Travel Application for Alpha Release
	Test REST APIs - For Bugs and Glitches
	Documentation
	Release Source Code, Documentation and GitHub Repository
Sprint Backlog	Travel Application Testing - Alpha Release
	REST APIs Testing
	Backend Server Test
	Test Application in AWS Platform
	Alpha Release
	Documentation
Sprint Review	Tested Travel Application (Locally, AWS Platform)
	Tested Backend Server for Bugs and Glitches
	Delivery of Project Content by following Agile Methodology
	Alpha Release (Deliver Source Code, Documentation and GitHub Repository)
Retrospective	Delay in Delivering APIs from Accommodation Team.
	No APIs were delivered from Taxi Service, Payment Service
	Tight timeline led to Integration of APIs (Working APIs for Alpha Release)
	Alpha Release includes Front End Development and Backend Development
	Under Estimation of Team Resources (Less Number of Resources), Strict Timeline.

Table 5. Sprint 4 details – Kanban Board



Alpha Release

Labels: Sprint 4, Release

Received: +
Start: 06/16/2022 25
Due: Today at 11:59 PM 27
End: Today at 10:30 PM 27

Creator
Members: P
Assignee
Requested By: Add
Assigned By: Add

Sort: -7.0625
List: Done

Description
Product Backlog

1. Test Travel Application for Alpha Release
2. Test REST APIs - For Bugs and Glitches
3. Documentation
4. Release Source Code, Documentation and GitHub Repository

Gaurav Honnavara Manjunath completed checklist Sprint Backlog of Alpha Release. 4 hours ago

Gaurav Honnavara Manjunath checked 6. Documentation in checklist Sprint Backlog of Alpha Release. 4 hours ago

Gaurav Honnavara Manjunath added checklist item to 'Sprint Backlog' in Alpha Release.
6. Documentation
4 hours ago

Gaurav Honnavara Manjunath completed checklist Sprint Backlog of Alpha Release. 4 hours ago

Gaurav Honnavara Manjunath checked 5. Alpha Release in checklist Sprint Backlog of Alpha Release. 4 hours ago

Gaurav Honnavara Manjunath uncompleted the checklist Sprint Backlog of Alpha Release. 4 hours ago

Gaurav Honnavara Manjunath edited comment undefined. 4 hours ago

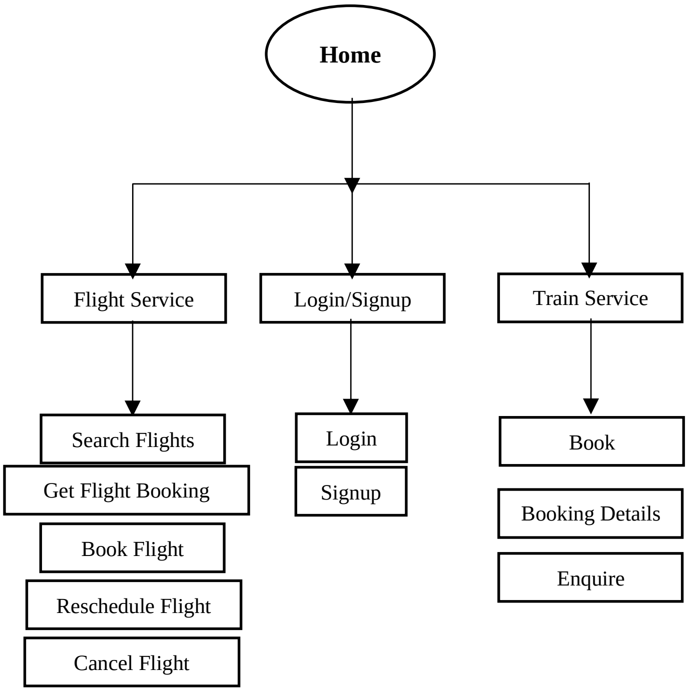
Fig 5. Sprint 4 details – Kanban Board

5.2 Programming Language & Tools

- **Platform**: Visual Studio Code
- **Language**: HTML, CSS, JS, Bootstrap
- **Cloud**: Amazon Web Service (EC2 Instance & S3 Storage)
- **Backend Server:** Apache24, phpMyAdmin, XAMPP Control Panel
- **Project Estimation** Tool: Kanban
- **Version Control:** GitHub Repository

5.3 Class Diagram

Class diagram of Travel Application as show below.



5.4 Front-End Interface (Home)

The Startup Page of Travel Application is as Shown below, the user can navigate across the page for login/Signup, go to train bookings Tab, Flight Booking Tab, Taxi booking Tab & Accommodation booking tab.

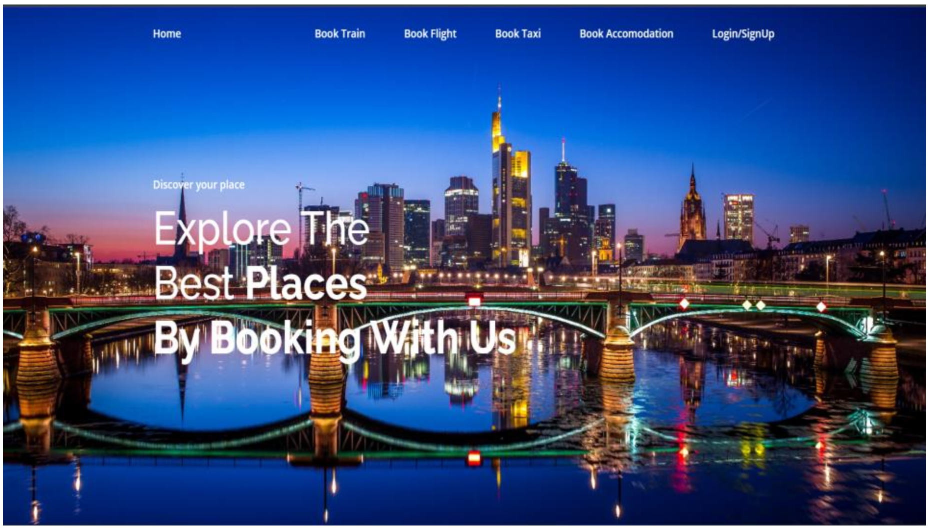
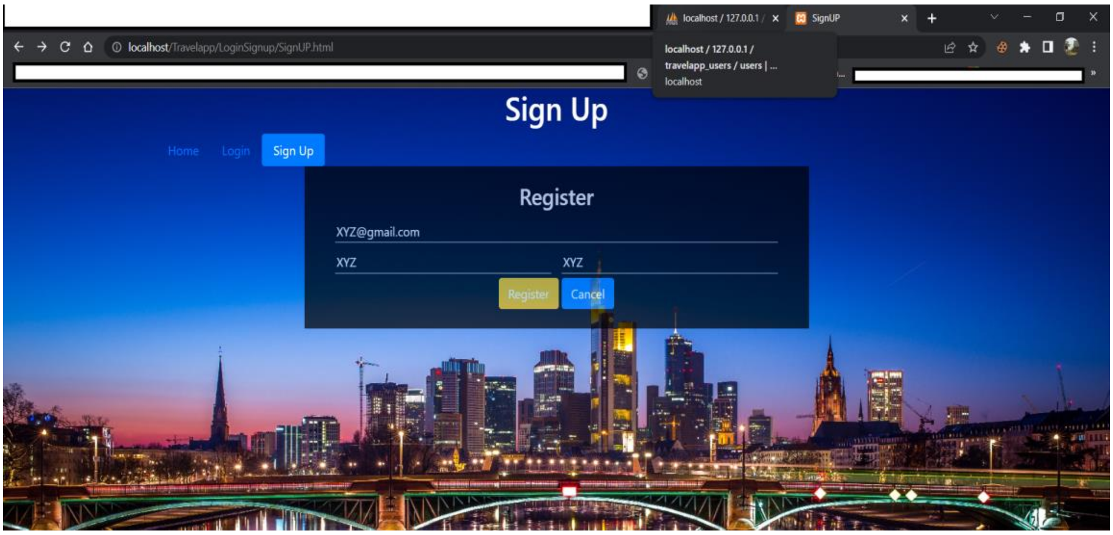


Fig 6. Travel App – Home

5.5 Front-End Interface (Login/Signup)

Travel Application allows the user to register with the page and login with User Credentials. Also, the User Credentials are stored in Backend MySQL server.



XYZ@gmail.com

XYZ XYZ

Fig 7. Travel App – Sign up

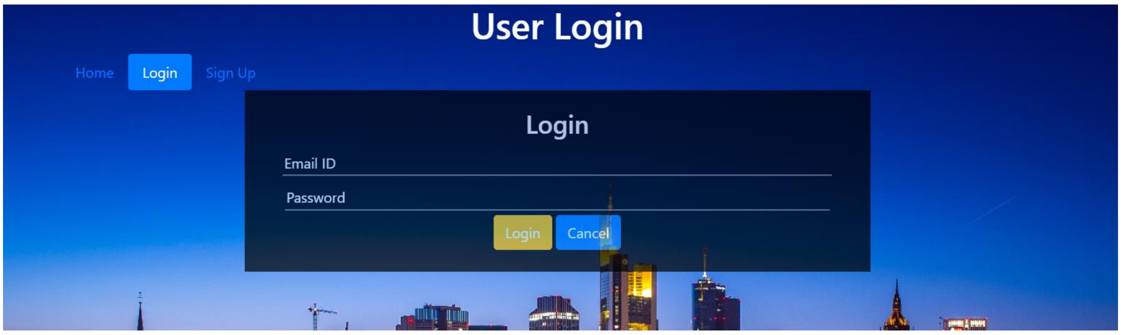
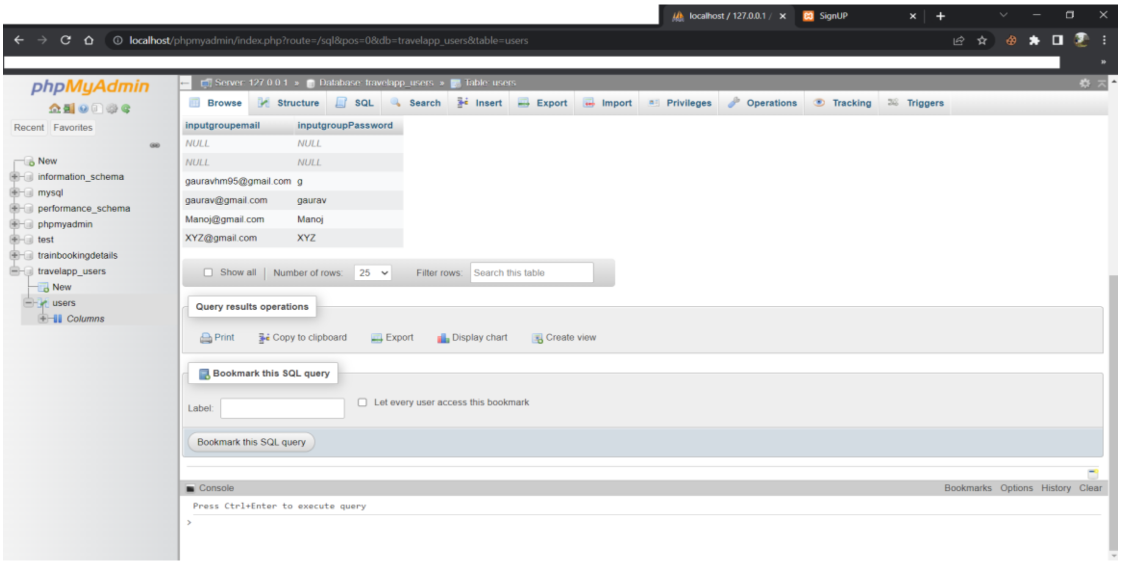


Fig 8. Travel App – Login



localhost/phpmyadmin/index.php?route=/sql&pos=0&db=travelapp_users&table=users

inputgroupemail	inputgroupPassword
NULL	NULL
NULL	NULL
gauravhm95@gmail.com	g
gaurav@gmail.com	gaurav
Manoj@gmail.com	Manoj
XYZ@gmail.com	XYZ

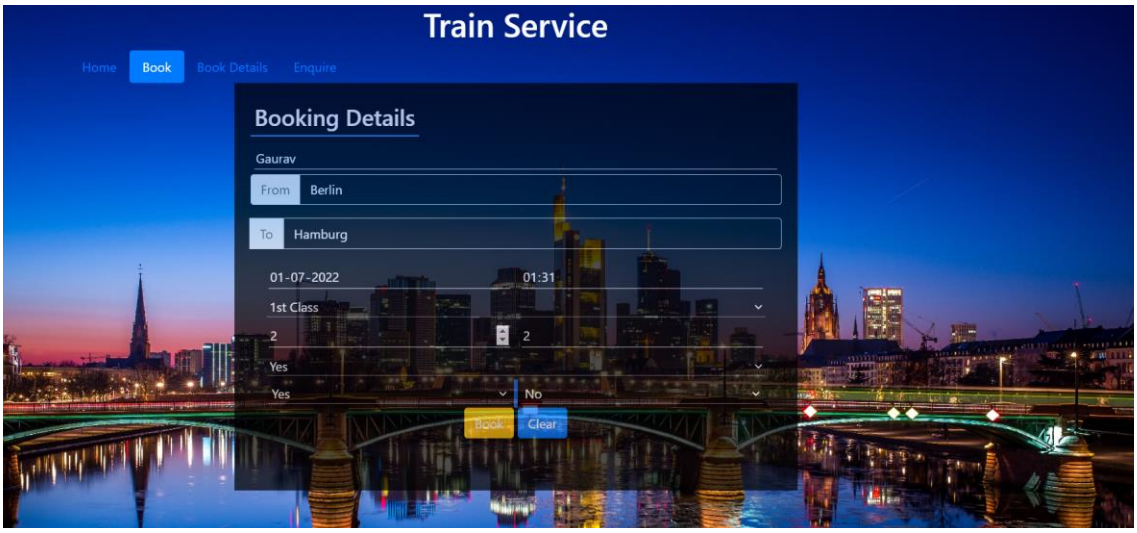
□ Show all | Number of rows: 25

```
Press Ctrl+Enter to execute query
>
```

Fig 9. Travel App – User Database (Backend)

5.6 Front-End Interface (Train Service)

The user can navigate through the Train Service tab and access sections to book, enquire about booking status, and check booking details and route details by accessing REST API.



Gaurav

From: Berlin

To: Hamburg

01-07-2022 01:31

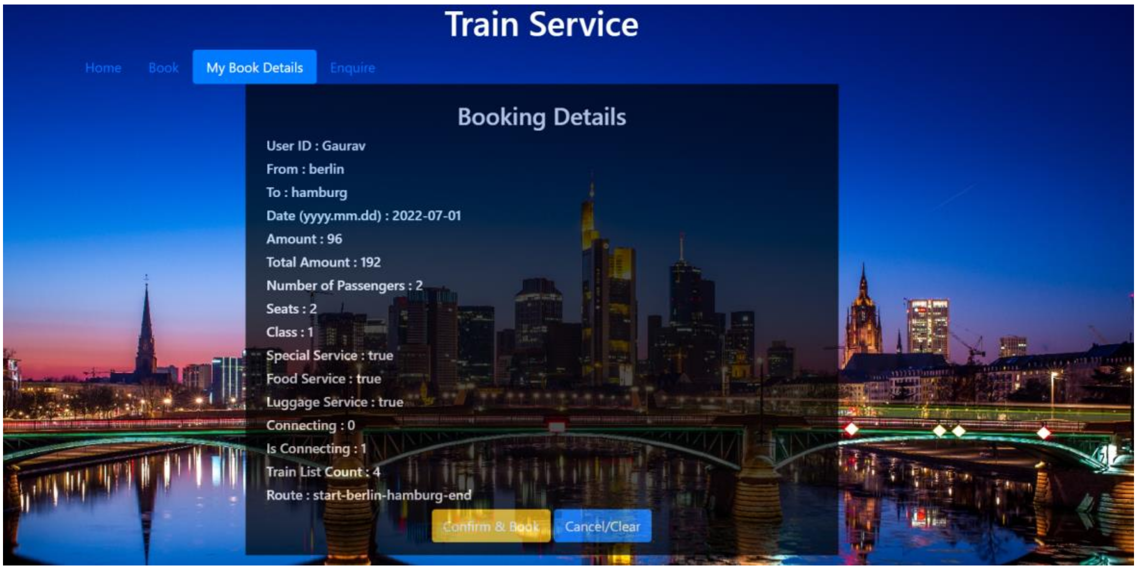
1st Class

2 2

Yes

Yes No

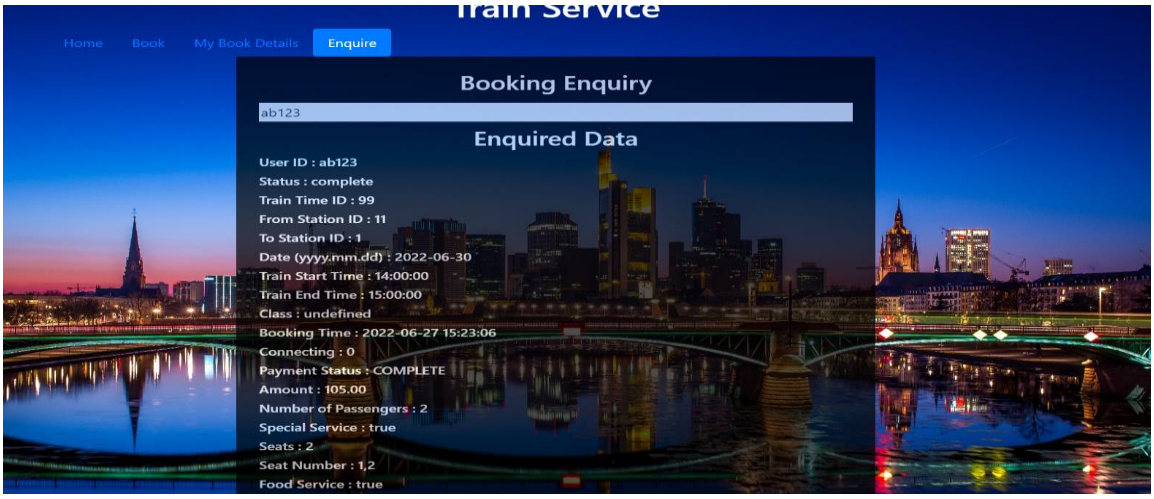
Fig 10. Travel App –Train Service (Book)



Booking Details

User ID : Gaurav
From : berlin
To : hamburg
Date (yyyy.mm.dd) : 2022-07-01
Amount : 96
Total Amount : 192
Number of Passengers : 2
Seats : 2
Class : 1
Special Service : true
Food Service : true
Luggage Service : true
Connecting : 0
Is Connecting : 1
Train List Count : 4
Route : start-berlin-hamburg-end

Fig 11. Travel App –Train Service (Book Details)



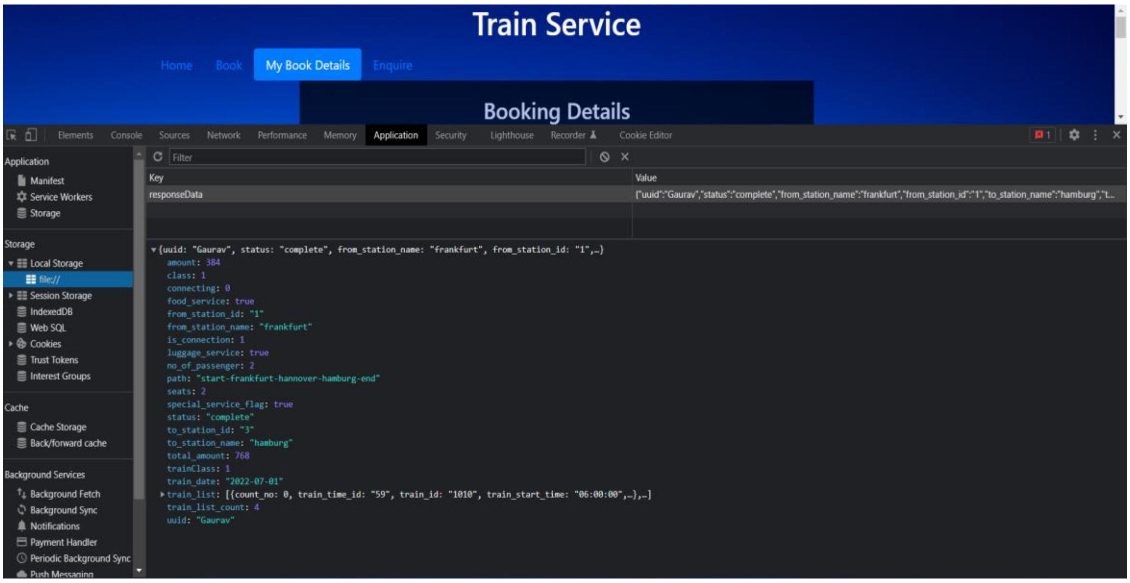
Booking Enquiry

ab123

Enquired Data

User ID : ab123
Status : complete
Train Time ID : 99
From Station ID : 11
To Station ID : 1
Date (yyyy.mm.dd) : 2022-06-30
Train Start Time : 14:00:00
Train End Time : 15:00:00
Class : undefined
Booking Time : 2022-06-27 15:23:06
Connecting : 0
Payment Status : COMPLETE
Amount : 105.00
Number of Passengers : 2
Special Service : true
Seats : 2
Seat Number : 1,2
Food Service : true

Fig 12. Travel App –Train Service (Enquiry)



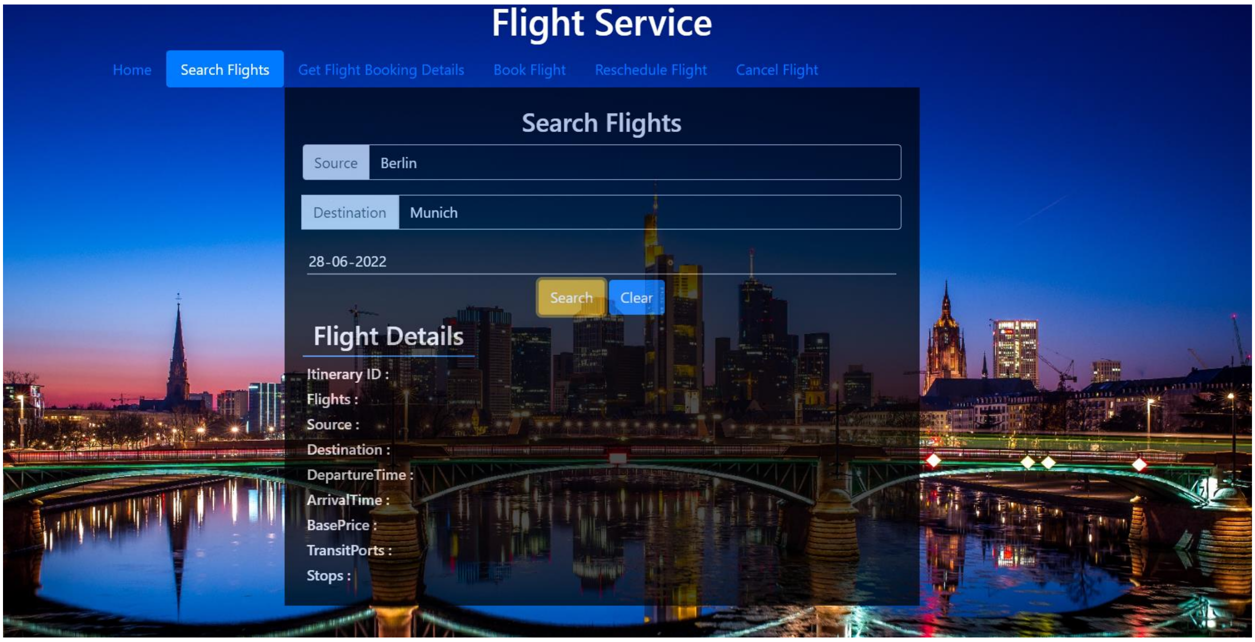
Key	Value
responseData	{"uuid":"Gaurav","status":"complete","from_station_name":"frankfurt","from_station_id":"1","to_station_name":"hamburg","t…

```
▾{uuid: "Gaurav", status: "complete", from_station_name: "frankfurt", from_station_id: "1",…}
   amount: 384
   class: 1
   connecting: 0
   food_service: true
   from_station_id: "1"
   from_station_name: "frankfurt"
   is_connection: 1
   luggage_service: true
   no_of_passenger: 2
   path: "start-frankfurt-hannover-hamburg-end"
   seats: 2
   special_service_flag: true
   status: "complete"
   to_station_id: "3"
   to_station_name: "hamburg"
   total_amount: 768
   trainClass: 1
   train_date: "2022-07-01"
  ▸train_list: [{count_no: 0, train_time_id: "59", train_id: "1010", train_start_time: "06:00:00",…},…]
   train_list_count: 4
   uuid: "Gaurav"
```

Fig 13. Travel App –Train Service (REST API GET)

5.7 Front-End Interface (Flight Service)

The user can navigate through the Flight Service tab and access sections to Search Flights, Get Flight Booking Details, Book Flight. Reschedule Flights and Cancel Booking by accessing REST API.



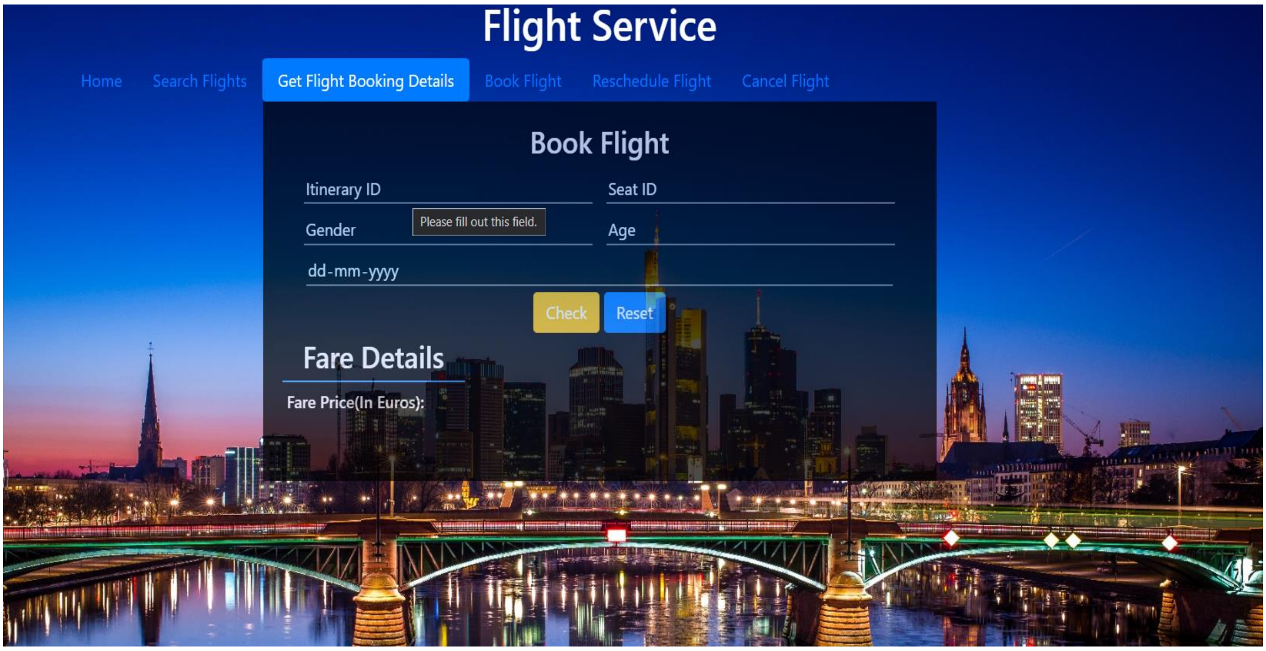
Source: Berlin

Destination: Munich

28-06-2022

Itinerary ID :
Flights :
Source :
Destination :
DepartureTime :
ArrivalTime :
BasePrice :
TransitPorts :
Stops :

Fig 14. Travel App –Flight Service (Search Flights)



Please fill out this field.

Fig 15. Travel App –Flight Service (Get Flight Booking Details)

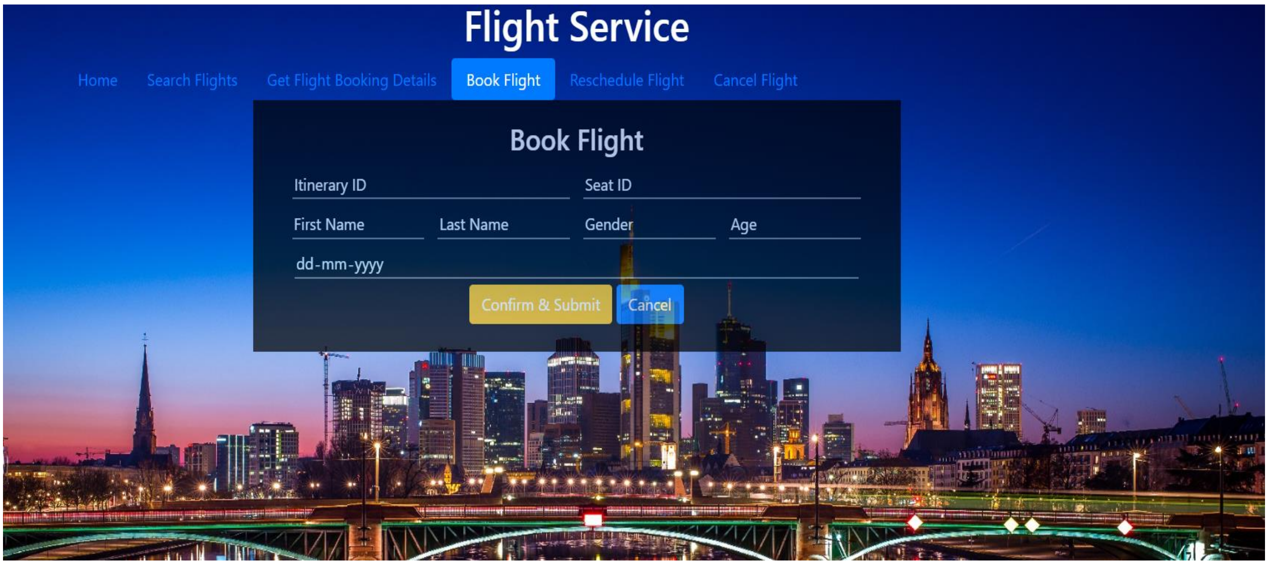


Fig 16. Travel App –Flight Service (Book Flight)

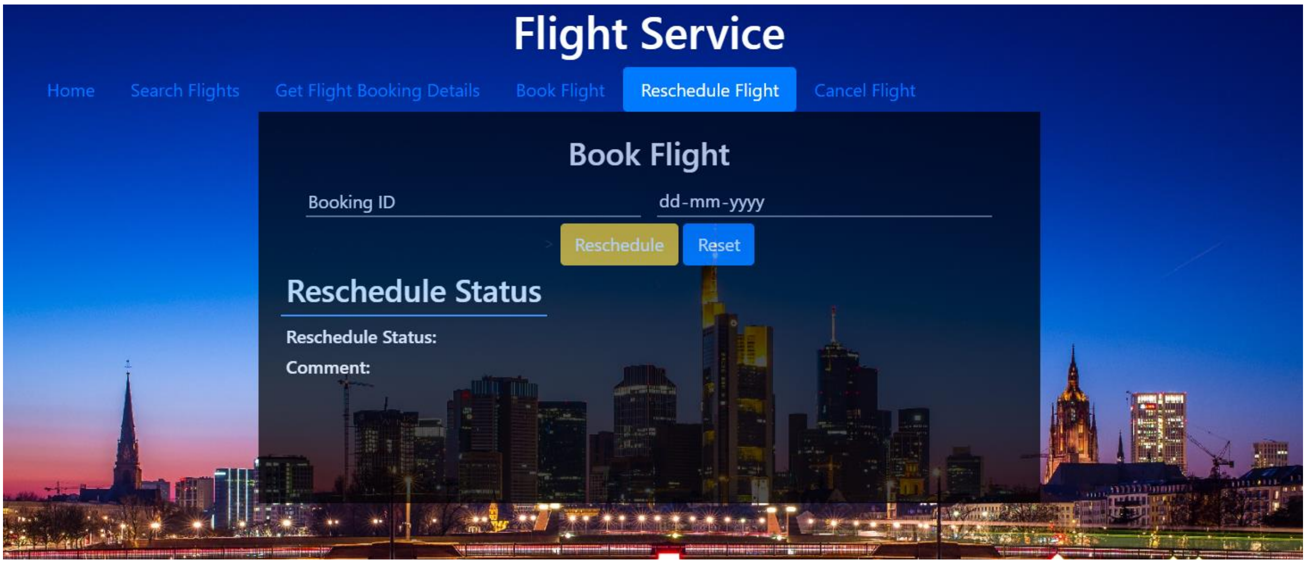


Fig 17. Travel App –Flight Service (Reschedule Flight)

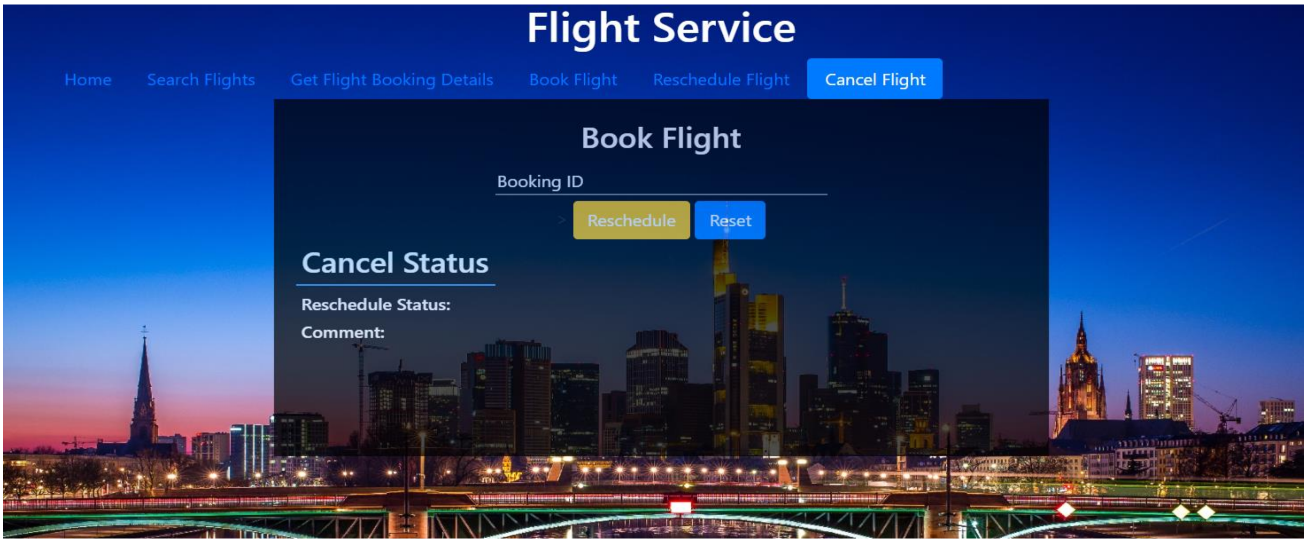
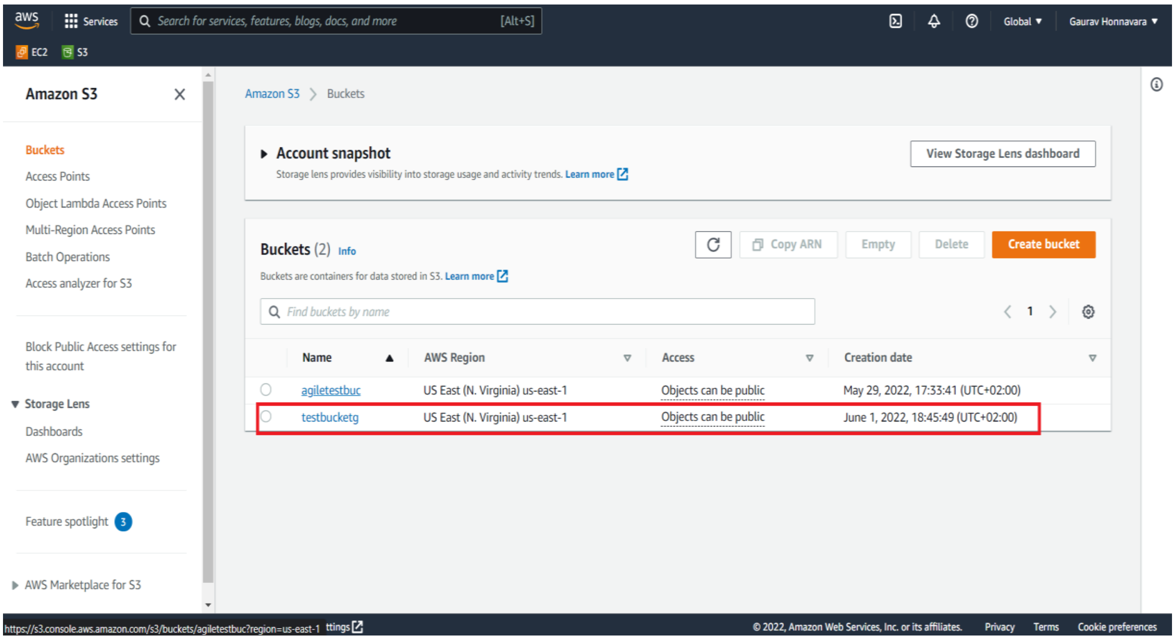


Fig 18. Travel App –Flight Service (Cancel Flight)

5.8 Amazon Web Services (Cloud Platform)

The Travel Application is deployed in Cloud by using Amazon web services Platform and hosted using EC2 Instance.

- Scale storage resources(S3) in AWS allows us to Store our Travel Application in Cloud. The Project is stored in cloud under “**testbucketg**” as shown below.



Account snapshot

Storage lens provides visibility into storage usage and activity trends. Learn more

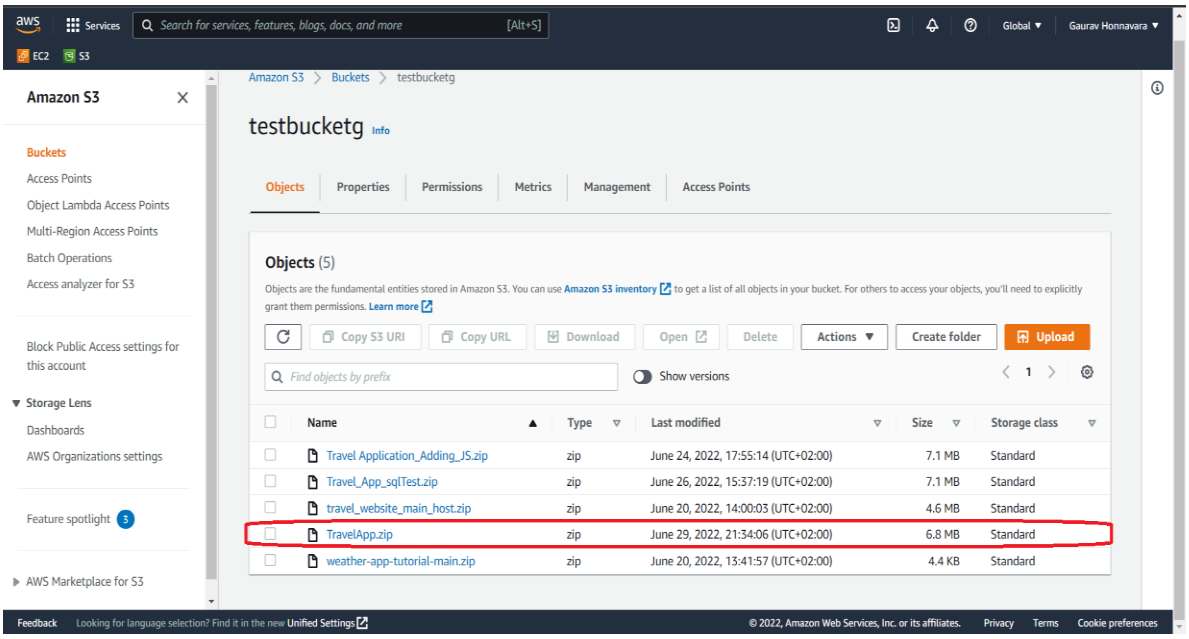
Buckets (2) Info

Buckets are containers for data stored in S3. Learn more

Name	AWS Region	Access	Creation date
agiletestbuc	US East (N. Virginia) us-east-1	Objects can be public	May 29, 2022, 17:33:41 (UTC+02:00)
testbucketg	US East (N. Virginia) us-east-1	Objects can be public	June 1, 2022, 18:45:49 (UTC+02:00)

https://s3.console.aws.amazon.com/s3/buckets/agiletestbuc?region=us-east-1 ttings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Fig 19. Travel App –S3 Storage (Cloud AWS)



testbucketg Info

Objects (5)

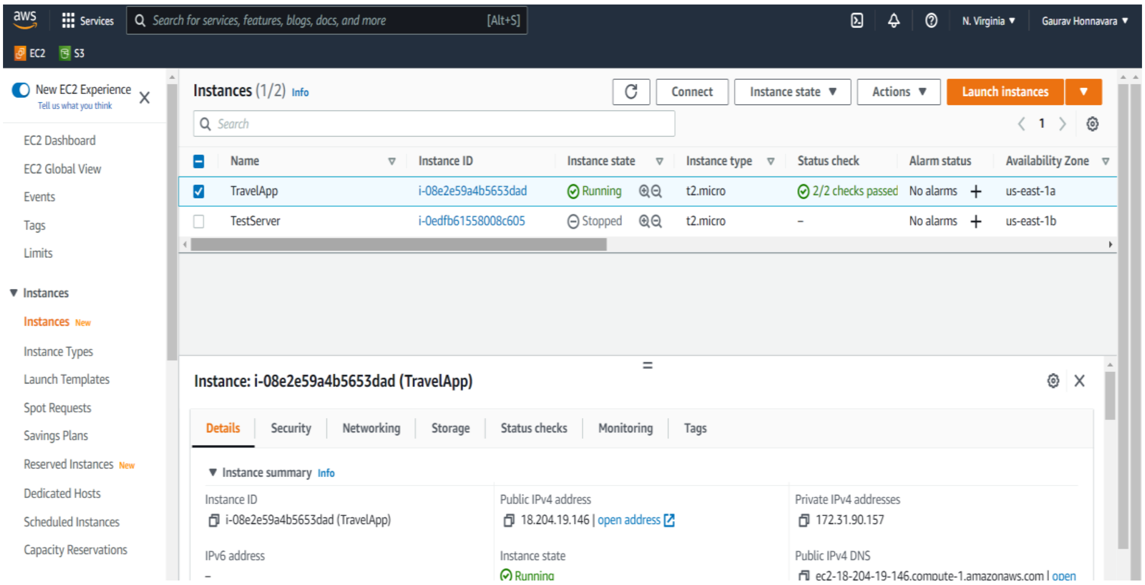
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Name	Type	Last modified	Size	Storage class
Travel Application_Adding_JS.zip	zip	June 24, 2022, 17:55:14 (UTC+02:00)	7.1 MB	Standard
Travel_App_sqlTest.zip	zip	June 26, 2022, 15:37:19 (UTC+02:00)	7.1 MB	Standard
travel_website_main_host.zip	zip	June 20, 2022, 14:00:03 (UTC+02:00)	4.6 MB	Standard
TravelApp.zip	zip	June 29, 2022, 21:34:06 (UTC+02:00)	6.8 MB	Standard
weather-app-tutorial-main.zip	zip	June 20, 2022, 13:41:57 (UTC+02:00)	4.4 KB	Standard

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Fig 20. Travel App –S3 Storage – TravelApp.Zip(Cloud AWS)

- Virtual server is created in Amazon's Elastic Compute Cloud (EC2) for running Travel Application on the Amazon Web Services (AWS) infrastructure.
- EC2 Instance – ‘**TravelApp’** is used to host our application – http://18.204.19.146/



Instances (1/2) Info

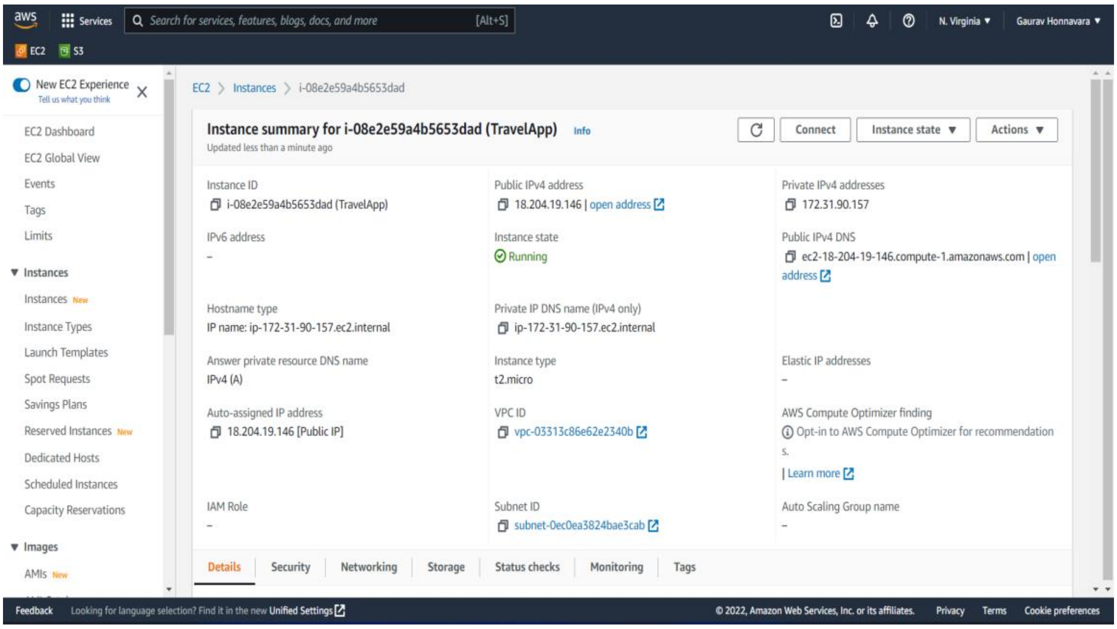
	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
☑	TravelApp	i-08e2e59a4b5653dad	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a
□	TestServer	i-0edfb61558008c605	Stopped	t2.micro	–	No alarms	us-east-1b

Instance: i-08e2e59a4b5653dad (TravelApp)

▼ Instance summary Info

Instance ID	Public IPv4 address	Private IPv4 addresses
i-08e2e59a4b5653dad (TravelApp)	18.204.19.146 | open address	172.31.90.157
IPv6 address	**Instance state**	**Public IPv4 DNS**
–	Running	ec2-18-204-19-146.compute-1.amazonaws.com | open

Fig 21. Travel App –EC2 Instance – TravelApp(Cloud AWS)



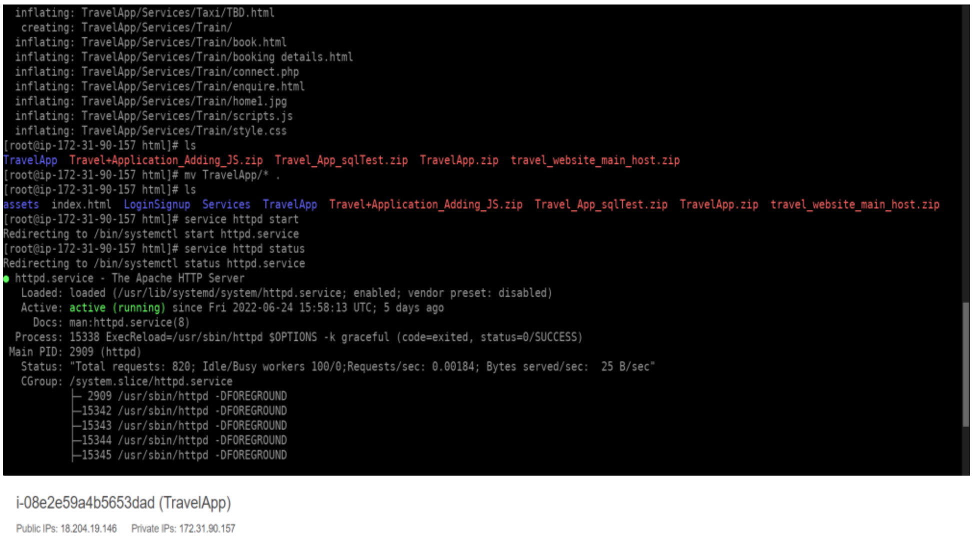
Instance summary for i-08e2e59a4b5653dad (TravelApp) Info

Updated less than a minute ago

Instance ID	Public IPv4 address	Private IPv4 addresses
i-08e2e59a4b5653dad (TravelApp)	18.204.19.146 | open address	172.31.90.157
IPv6 address	**Instance state**	**Public IPv4 DNS**
–	Running	ec2-18-204-19-146.compute-1.amazonaws.com | open address
Hostname type	**Private IP DNS name (IPv4 only)**	
IP name: ip-172-31-90-157.ec2.internal	ip-172-31-90-157.ec2.internal	
Answer private resource DNS name	**Instance type**	**Elastic IP addresses**
IPv4 (A)	t2.micro	–
Auto-assigned IP address	**VPC ID**	**AWS Compute Optimizer finding**
18.204.19.146 [Public IP]	vpc-03313c86e62e2340b	Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role	**Subnet ID**	**Auto Scaling Group name**
–	subnet-0ec0ea3824bae3cab	–

Fig 22. Travel App –EC2 Instance – TravelApp(Cloud AWS)

- Starting Travel Application using httpd Service



```
  inflating: TravelApp/Services/Taxi/TBD.html
   creating: TravelApp/Services/Train/
  inflating: TravelApp/Services/Train/book.html
  inflating: TravelApp/Services/Train/booking details.html
  inflating: TravelApp/Services/Train/connect.php
  inflating: TravelApp/Services/Train/enquire.html
  inflating: TravelApp/Services/Train/home1.jpg
  inflating: TravelApp/Services/Train/scripts.js
  inflating: TravelApp/Services/Train/style.css
[root@ip-172-31-90-157 html]# ls
TravelApp  Travel+Application_Adding_JS.zip  Travel_App_sqlTest.zip  TravelApp.zip  travel_website_main_host.zip
[root@ip-172-31-90-157 html]# mv TravelApp/* .
[root@ip-172-31-90-157 html]# ls
assets  index.html  LoginSignup  Services  TravelApp  Travel+Application_Adding_JS.zip  Travel_App_sqlTest.zip  TravelApp.zip  travel_website_main_host.zip
[root@ip-172-31-90-157 html]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-90-157 html]# service httpd status
Redirecting to /bin/systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Fri 2022-06-24 15:58:13 UTC; 5 days ago
     Docs: man:httpd.service(8)
  Process: 15338 ExecReload=/usr/sbin/httpd $OPTIONS -k graceful (code=exited, status=0/SUCCESS)
 Main PID: 2909 (httpd)
   Status: "Total requests: 820; Idle/Busy workers 100/0;Requests/sec: 0.00184; Bytes served/sec:  25 B/sec"
   CGroup: /system.slice/httpd.service
           ├─ 2909 /usr/sbin/httpd -DFOREGROUND
           ├─15342 /usr/sbin/httpd -DFOREGROUND
           ├─15343 /usr/sbin/httpd -DFOREGROUND
           ├─15344 /usr/sbin/httpd -DFOREGROUND
           └─15345 /usr/sbin/httpd -DFOREGROUND
```

i-08e2e59a4b5653dad (TravelApp)

Public IPs: 18.204.19.146   Private IPs: 172.31.90.157

Fig 23. Travel App –Starting httpd service – TravelApp(Cloud AWS)

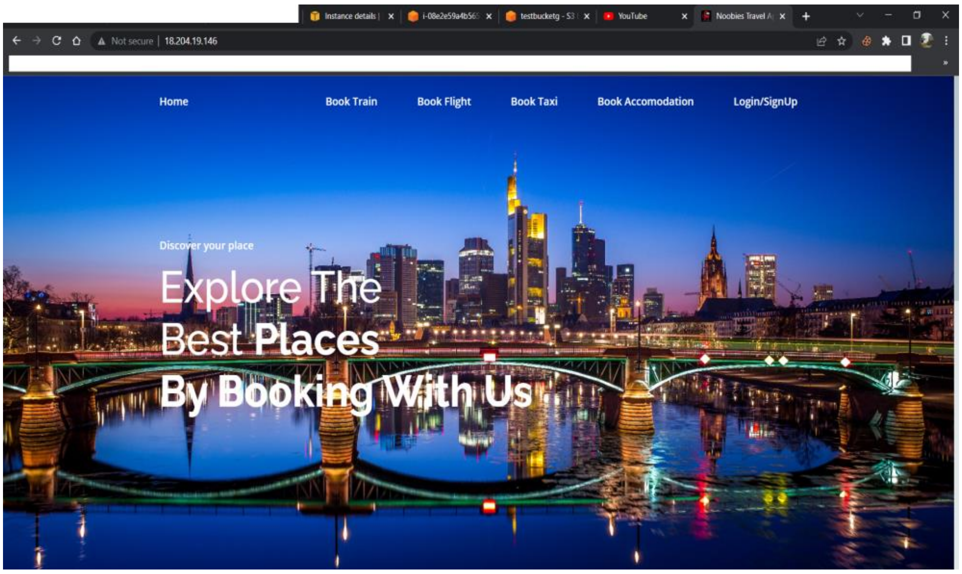
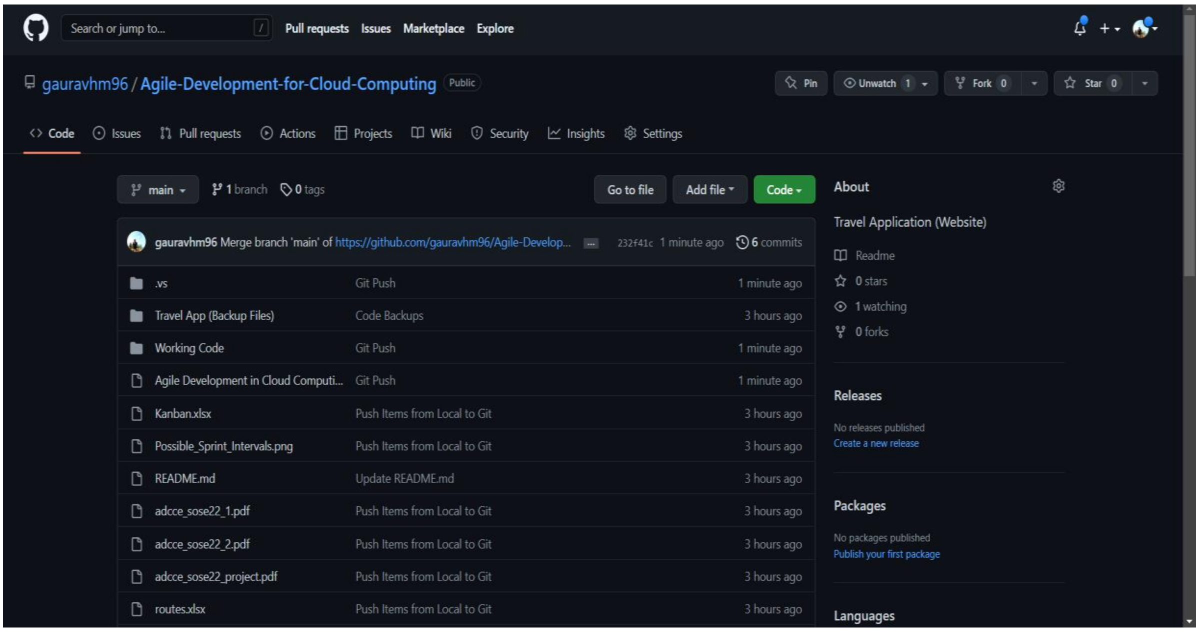


Fig 24. Travel App – (Cloud AWS)

5.9 Version Control

The Team worked on development and maintained version control using GitHub Repository.

GitHub Repository - https://github.com/gauravhm96/Agile-Development-for-Cloud-Computing



Travel Application (Website)

No releases published

No packages published

gauravhm96 Merge branch 'main' of https://github.com/gauravhm96/Agile-Develop... 232f41c 1 minute ago 6 commits

.vs	Git Push	1 minute ago
Travel App (Backup Files)	Code Backups	3 hours ago
Working Code	Git Push	1 minute ago
Agile Development in Cloud Computi...	Git Push	1 minute ago
Kanban.xlsx	Push Items from Local to Git	3 hours ago
Possible_Sprint_Intervals.png	Push Items from Local to Git	3 hours ago
README.md	Update README.md	3 hours ago
adcce_sose22_1.pdf	Push Items from Local to Git	3 hours ago
adcce_sose22_2.pdf	Push Items from Local to Git	3 hours ago
adcce_sose22_project.pdf	Push Items from Local to Git	3 hours ago
routes.xlsx	Push Items from Local to Git	3 hours ago

Fig 25. Git Repository

6 Agile best practices which have been incorporated in our project

'Agile' has offered an effective replacement to the conventional waterfall model of project management. Agile practices focus more on collaboration with the customer and/or end user by continuously delivering through self-organisation or cross functional team effort, the necessary requirements based on adaptive planning, effective implementation, communication, early delivery etc. This methodology was required to assist companies become more flexible, responsive, and adaptable to change. As customer is regarded as the centrepiece of any business model, the following agile principles have been thoroughly put into effect when implementing our project:

- One of the top priorities of the Agile approach is to aim for customer satisfaction through prompt and continual delivery of value-added products. We had call with our customer every week to discuss the progress and continual delivery of the service customer agrees too.

- The agile approach must embrace changes, even at later stages of development. Change is considered as a valuable agent offering a competitive edge to the customers. In our project we had many changes during our sprints, and we were able to manage the change even during our last sprints to make sure we deliver the best to our customer.

- Working software should be delivered to the customers for any feedback or suggestions at a periodic interval of weeks or months, preferably in shorter time spans. We have delivered our software and any suggestion or feedback from our customer is welcome and continuously we will work on it. Ultimately customer satisfaction is our priority.

- The business management and system developers should work collaboratively throughout the project cycle. We had calls for discussion of the progress made through our sprints and, we had the Kanban board to track the progress. We also collaborated via emails and chats for the updates.

- The agile approach believes in assigning projects to motivated individuals and provide the individuals with a good working environment and support system by inculcating trust in them to get the work done. We organise the tasks for every sprint and divided the task among our team for the best results.

- Agile Values and Principles are about fostering face-to-face discussions and promoting them as an effective means of communication to and within the team of developers. Face-to-face communication can be seen as a direct approach to an idea by the means of discussions. The human interaction helps in decoding the messages conveyed and ensures that ideas get interpreted correctly. However, given the current pandemic situation it was quite meaningful that these interactions be held via online platforms such as Zoom, Microsoft Teams. The meetings were held virtual and enabled us to have meaningful discussions using the Video chat and Share screen feature. In fact, the share screen feature helped the team solve a lot of issues, review the code and fix bugs in a more efficient manner than what face-to-face meetings could have offered.

- Working software is one of the primary means of measuring system progress in agile. A weekly basis is more desirable to achieve the stated level of expectation. This was incorporated in the project by working on tasks defined in the Kanban Scrum board on a weekly basis. The attributes of the customer’s needs were considered and worked on by individual members of the team which reported by the end of the week the corresponding implemented controls.

- The agile approach aims at facilitating consistent and sustainable progress in system development. The stakeholders are required to maintain steady and continual velocity in product development. As stated before, Agile focuses more on collaborative effort. Hence, individual contributions in the respective field with efficient communication provides a seamless delivery. The idea is to get the individuals who produce value and those who plan or sell it to work together. This ensures that the internal cooperation is effortless, and the process performance improves. This principle played a huge part in the development of the project as there was effective communication among the team members when sorting out the tasks, assigning priorities etc. Certain tasks in our project depends upon the results of the other team's output. The constant updates from team members helped seamlessly achieve the completion of tasks ensuring collaborative effort in place.

- Self-organized teams help in bringing good structures, requirements, and developments. The agile teams are proactive, which enables them to overcome any hurdles and minimize errors to deliver excellence.

- Agile teams continually upgrade themselves and aim to become more productive by adapting to changes effectively. The development teams constantly mold and are at peace with the changing external conditions. It enables continual improvisation to deliver value. Periodic feedback from the customers enables the development team to attain it.

7. Application of Push and pull principle

- Pushing principle in agile means assigning work to a developer.
- Pulling in agile means developer grabbing their next task once they have bandwidth to work.
- In our project we have applied the pushing principle initially to assign who will be working on which component/flow or part in the project.
- Later we have applied the pulling principle where when someone completes their assigned task, they will take up a new task or help their teammates in completing the tasks of the sprint backlog.

8. Measuring the complexity of our tasks

- Task complexity has been recognized as an important task characteristic that influences the performance of our project.
- The complexity of a task can be represented in story points assigned to the task.
- We measured the complexity of our task by measuring the amount of task need to be done in our respective sprints and the available resource we have in our team.
- Difficult part during the assignment part was the estimation of the time we need to complete the task since we had limited resources in our team and we had many tasks to be completed in order to deliver the software on time.

9. Collaboration with the customer.

- Our project needs the active collaboration with the customer to provide the updates regarding our sprints and with other teams for us to understand the API model they are work with.
- Collaboration for our travel app with customer was done on weekly basis via zoom calls. Progress of the sprints were discussed, and mistakes of the sprints were focused on the call

so that the mistakes were identified in earlier stages reducing the risk of the mistakes on our next sprints.

- According to Agile methodology, it’s better to “Fail fast, fail cheap, fail early”. Hence, collaboration and active communication of our progress helped a lot in our project to move in the right direction.

10. Necessity of having the roles Product Owner and Scrum Master.

- Initially during the planning stage, that is sprint 1 was completely started without any specific roles which led to misunderstanding of the assignment. As a result, tasks were allocated to each one based on each one's voluntariness and not based on strong design logic.
- Hence, we started our project with C#, and we could complete the task during our sprint 1 and we got to know the future problem that might arise due to this.
- These issues were rectified in sprint 2, where one team member took up the responsibility of being the product owner and the scrum master and assigning the work based on design logic. The product owner then explained the logic to each of the team members to use the template and edit it according to the task assigned to them.
- Later, all our sprints went very well as planned.

11. Conclusion

Agile was born to simplify the lives of software developers, testers, and the organizations. Learning Agile signals that you need to be flexible and shift your creativity-discipline equilibrium more towards the creative side of Agile development. Hence Agile is a best method over the Traditional methods.

Since we are hosting our website on cloud environment platforms it is very easy to scalable further according to the customer needs in the future. Although few times it is hard to estimate the time takes for the task and few backlogs will be carried to the next sprints which is the actual drawback of the agile.

Development of Travel Application was done by incorporating Agile Methodology, and the team successfully performed assigned tasks and responsibilities in each sprint runs.
Though there were under estimations done in project planning, the team pushed to complete the tasks and deliver it to the client before the deadline.