

Pick Up

L u c a & P a u l & P h i l i p

Team NO.22



1

PART ONE

Project Intro



2

PART TWO

Structure



3

PART THREE

Core Function





Part 1 Project Introduction

Why We Come Up with PickUp?





Our Goal:

Help people find fellows who have the same team up needs as they do through our app.



What You Can Do on Our App

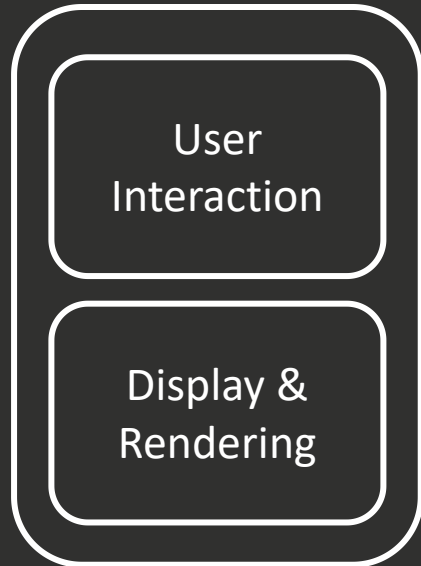
- 1.Join others' activity.
- 2.Post your own activity.



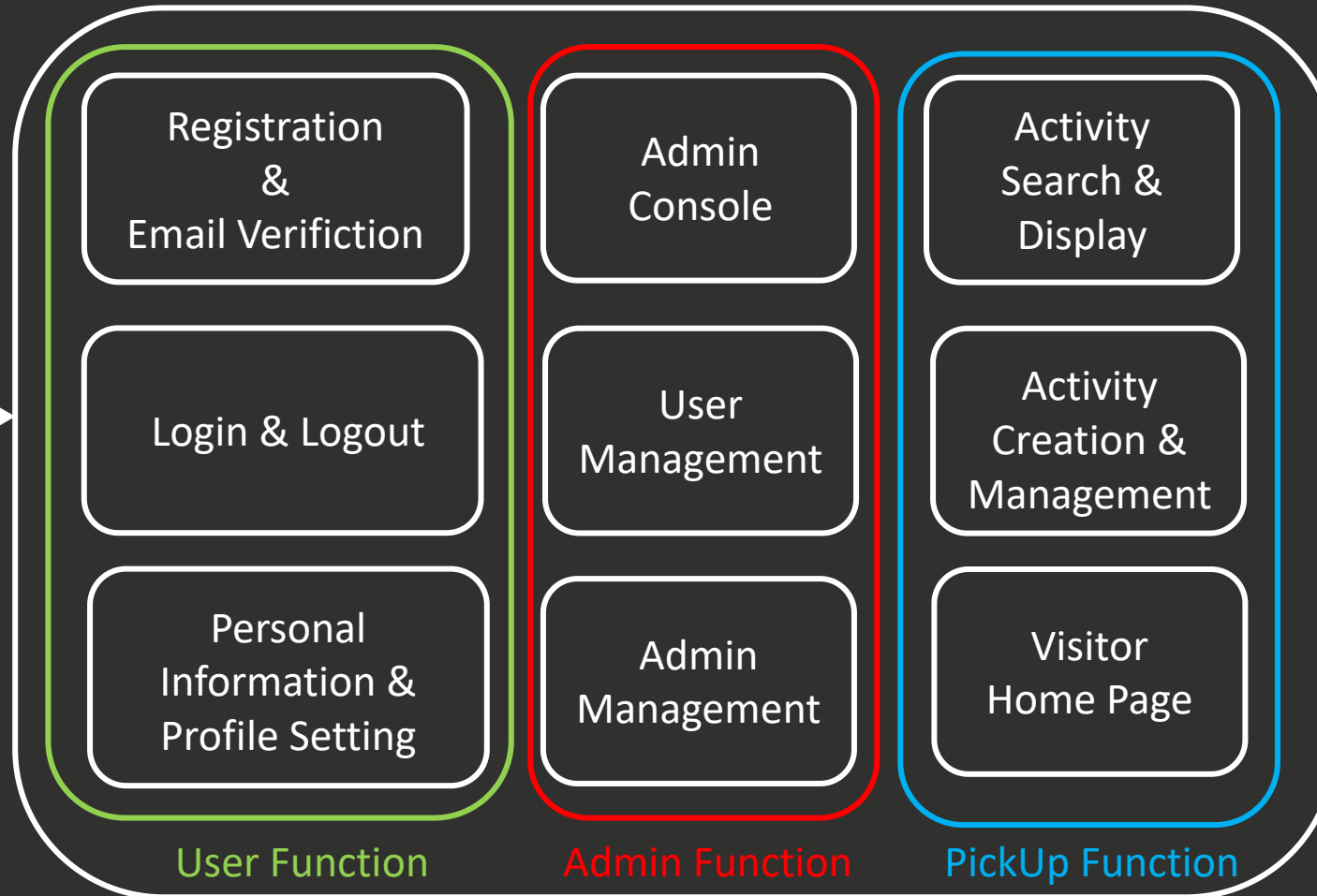
Part 2: Project Structure & Implementation Overview

Main Structure

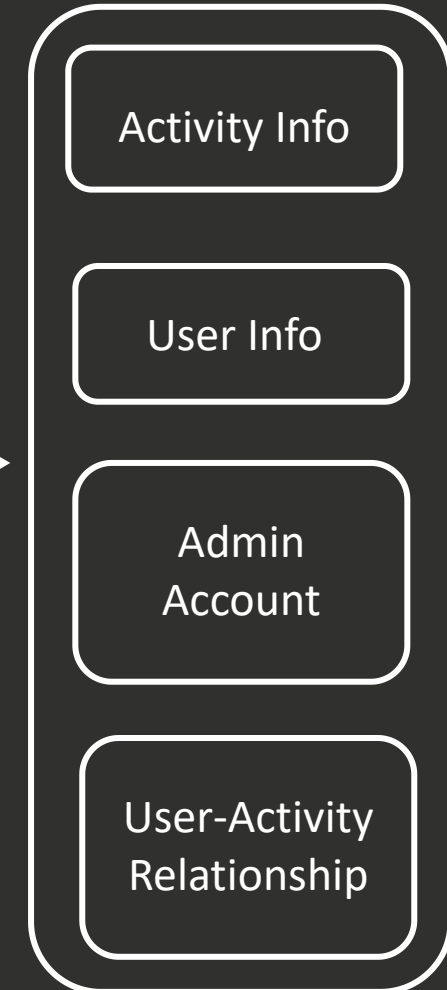
Client



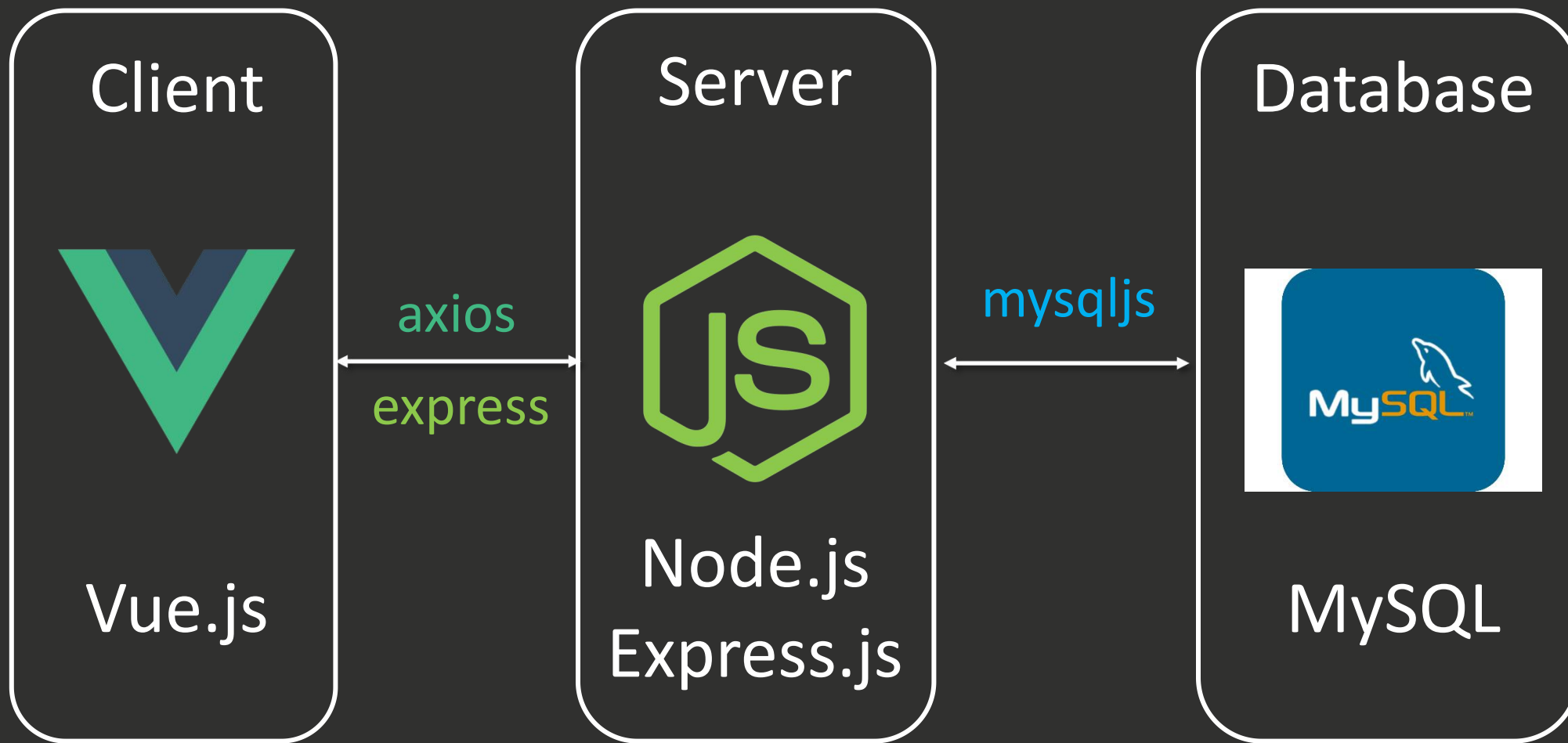
Server



Database



Technology Stack



Implementation Specification

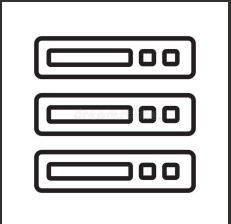


Front-end Framework: Vue.js

- Version: v3.2.13
- Development Server
 - Vue CLI

Front-end UI Component Library: Bootstrap

- Version: v5.1.3



Back-end Framework: Express.js

- Version: v4.17.3

JavaScript Runtime Environment: Node.js

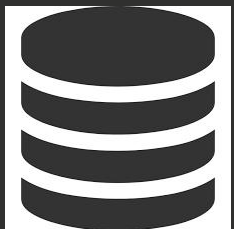
- Version: v14.18.0



Amazon
EC2

Cloud Deployment:

- Front-end & Back-end Hosting
 - Amazon Web Service Elastic Compute Cloud (EC2)



Database: MySQL

- Version: v8.0.27

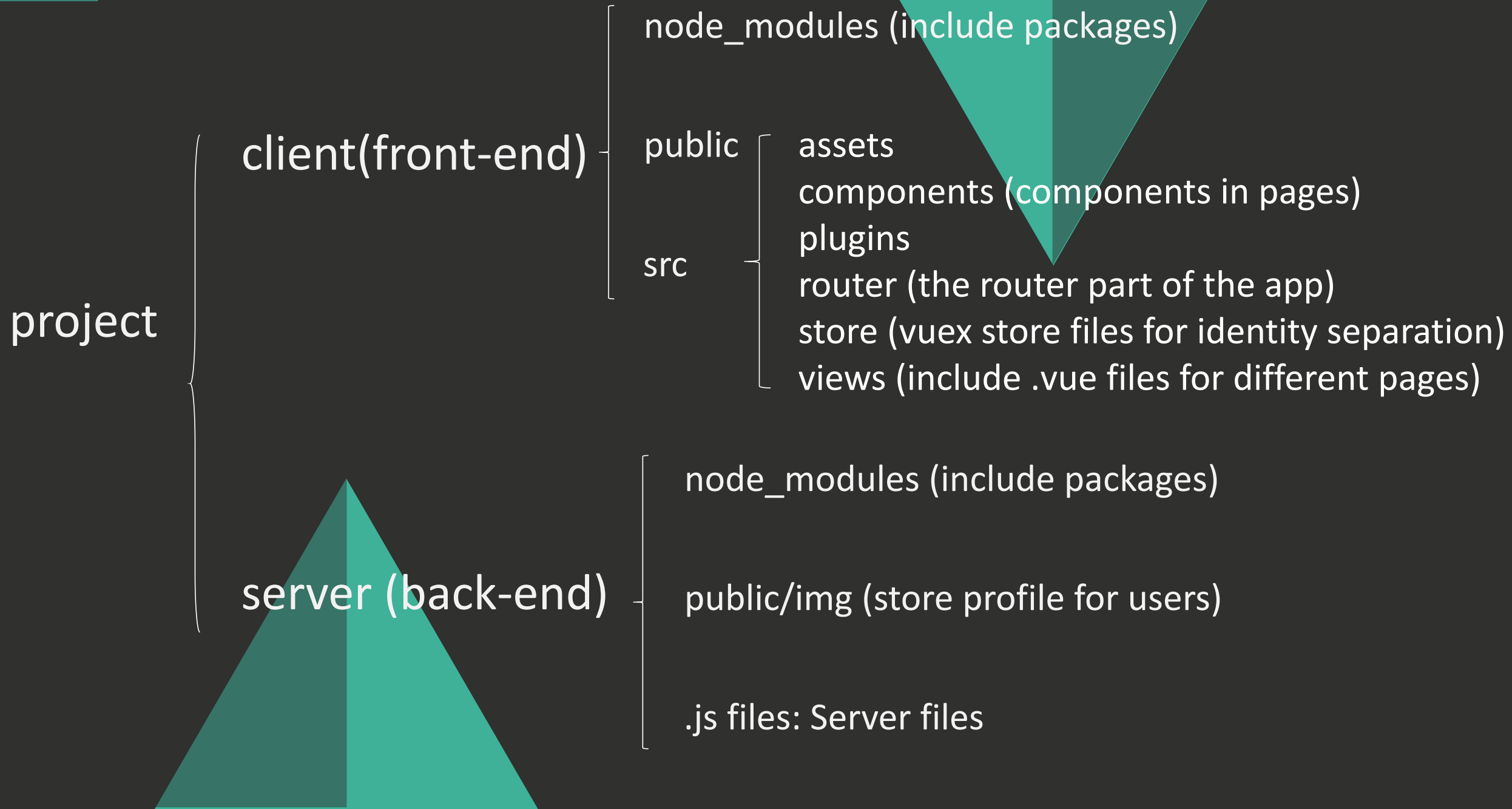


Amazon RDS

Cloud Deployment

- Remote Database
 - Amazon Web Service Relational Database Service (RDS)

File Structure



Tools

Back-end API Testing Tool:

- POSTMAN



JavaScript Code Analyzer

- ESLint



Version Control

- Git & Github





Part 3 Core Functions





Sign Up & Log In

- Function
 - User signup
 - User Login & Logout
 - Email verification
 - Identification (Admin user ; normal user)
- Package
 - nodemailer
 - vuex (Inter-component Data Management)



Home Page & Post Activity

- Function
 - Show all activity
 - Sort activity (according to create time)
 - Search activity (by type / by title/ by date)
 - Join activity
 - Post new activity



Manage Your Activity

- Function
 - Show all participated activity
 - Show activity info
 - Manage activity member
(Manager/member - quit / terminate / delete)



Setting & Personal Information

- Function
 - Reset user information (contact, password, user name, gender, personal introduction, profile)
 - List user information



Admin Console

- Function
 - List all users
 - Reset user password
 - Delete user
 - Create Admin account
 - Reset Admin account password
 - Delete Admin account



Routing

- Function
 - Redirect between pages
- Package
 - vue-router



Part 4 Highlight

High Lights

- Project Specific Function Fulfilling Daily Life Needs

PickUp

HomeActivity ManagementPersonal Center

Logout

Welcome to PickUp, Hope you can find friends here.

Now is: 2022/4/26 09:42:49 Tuesday

+ Post a new event

typesearch

Please select oneSort

26 Apr 2022search

title: Go hiking

type: Travel

time: 2022-04-30

location: ShenXian Lake

title: Play Basketball

type: Sports

time: 2022-04-30

location: Diligntia Sports

title: Let's eat healthy food

type: Meals

time: 2022-05-01

location: Muse Canteen

title: Go to BaoAn airport

type: Carpool

time: 2022-05-01

location: BaoAn District

title: Buy tasty bread

type: Shop online

time: 2022-04-28

location: Diligntia College

title: Run!

type: Sports

time: 2022-04-28

location: University Sports Hall

previous

1

next

go to:

PickUp

HomeActivity ManagementPersonal Center

Logout

Post a New Event

Post

Activity Type

SportsMealsTravelShop onlineCarpool

Activity Title

Activity Time

26 Apr 2022

-

00

:

00

No. of Members

Two

Three

Four

More

Activity Description

Activity Location

High Lights

- Cloud Deployment on Amazon Web Service
- Elastic Cloud Computing
- Relational Database Service

The screenshot displays the AWS Management Console interface for an EC2 instance. The left sidebar shows navigation options like EC2 Dashboard, Global View, Events, Tags, Limits, and Instances. The main content area shows the 'Instance summary' for 'i-072abadb6168e9015 (test-server)'. Key details include: Instance ID, Public IPv4 address (54.172.232.138), Private IPv4 addresses (172.31.80.54), Instance state (Running), Hostname type, IP name, Instance type (t2.micro), VPC ID, Subnet ID, AMI ID, AMI name, AMI location, Launch time, and Monitoring status (disabled).

The screenshot displays the AWS Management Console interface for an Amazon RDS database instance. The left sidebar shows navigation options like Amazon RDS, Databases, and database-2. The main content area shows the 'database-2' instance details. Key details include: DB identifier, CPU usage (5.74%), Status (Available), Engine (MySQL Community), Role, Instance, Connectivity & security, Monitoring, Logs & events, Configuration, Maintenance & backups, Tags, Endpoint & port, Networking, and Security.

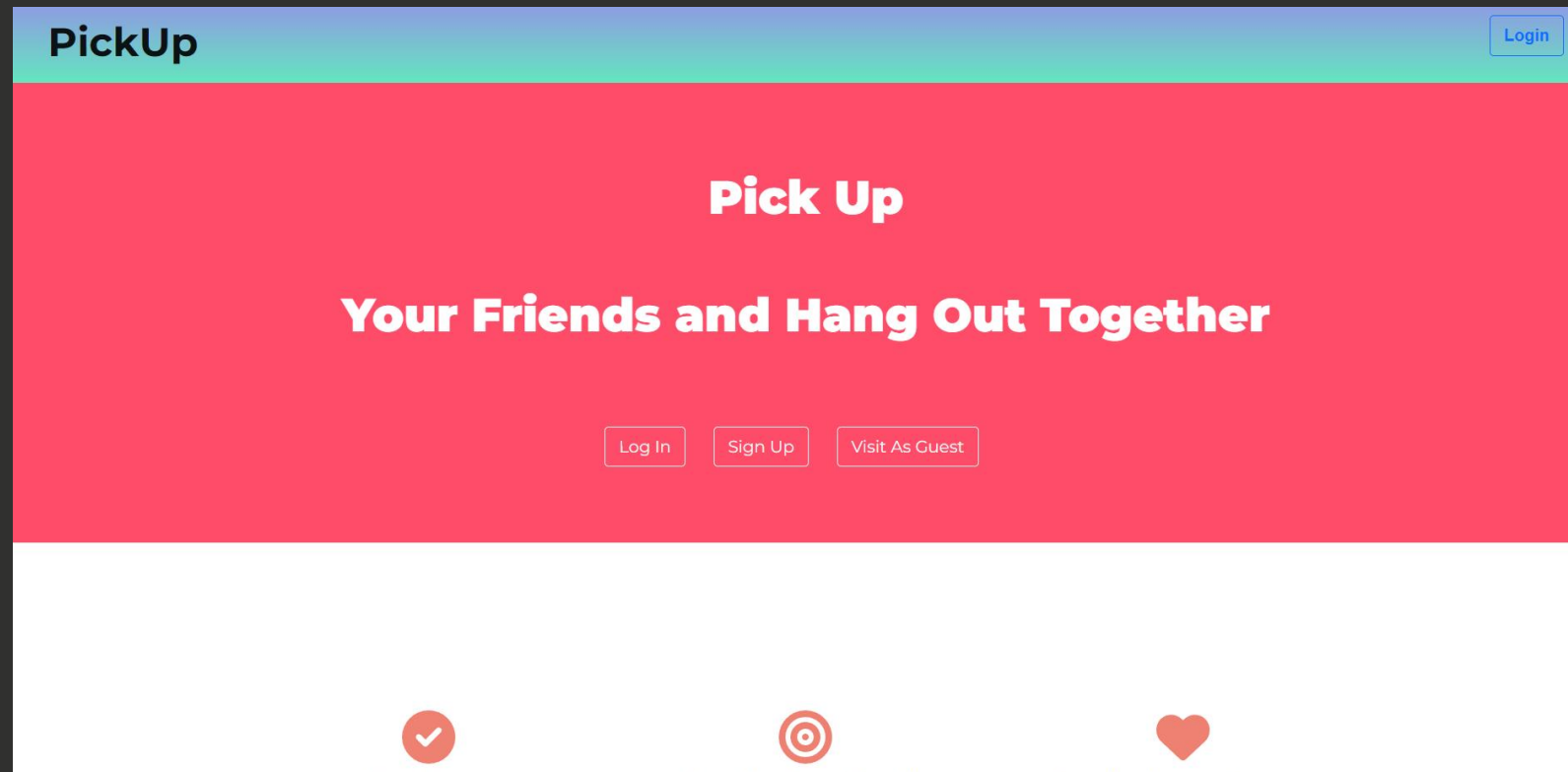


High Lights

- Cloud Deployment on Amazon Web Service
- You Can Play Around with Our App at <http://54.172.232.138/>

High Lights

- Pretty UI and Color Selection





THANKS

Function List Part 1

- User register/ User login (by Luca)
- User information editing (by Luca)
 - Change email
 - Change password
- Email Verification (by Luca)
 - Send verification code to user email and server
 - Verify the two code
 - Expire within 60 seconds
 - After verification succeed, store user account info into the database



Function List Part 2

- Create Activity and Join Activity (by Paul)
 - Time: minimum time unit every 15/30 minutes
 - Join activities posted by others
- Activity Management (by Philip)
 - Edit existing activity information
 - Add/delete participants
 - Change manager: initiator is the default manager
- Activity Search (by Paul)
 - Given tag
 - Create Time (Ascending/Descending)
- Admin Account (by Philip)
 - Manage other user



- *User register/ User login (by Luca)*



Front-End

In LoginView.vue:
userLogin();

In RegisterView.vue:
postMessage();



Back-End

In index.js:
app.post("/register", function(req, res))
app.post("/login", function(req, res))

- *User information editing (by Luca)*



Front-End

In PersonalInfoView.vue:

```
getUserInfo();
```

In SettingView.vue:

```
updatePassword() & updateUserName() & updateContact() & updateBio()
```



Back-End

In profileAndSetting.js:

```
app.get('/personalInfo', function(req,res))  
app.post("/setting/password", function(req, res))  
app.post("/setting/name", function(req, res))  
app.post("/setting/contact", function(req, res))  
app.post("/setting/bio", function(req, res))
```

- *Email Verification (by Luca)*



Front-End

In RegisterView.vue:
postMessage();



Back-End

In index.js:
app.get("/verify", function(req, res))

- *Create Activity and Join Activity (by Paul)*



Front-End

In ActivityCreationView.vue:

```
postActivity();  
tryJoin()
```



Back-End

In homeServer.js:

```
app.post("/postActivity", function(req, res))  
app.post("/joinActivity", function(req, res))
```

•Activity Management (by Philip)



Front-End

In ActivityCreationView.vue:

```
deleteMember()  
changeManager()  
terminateActivity()
```



Back-End

In homeServer.js:

```
app.post("/deleteMember", function(req, res))  
app.post("/changeManager", function(req, res))  
app.post("/terminateActivity", function(req, res))
```

•Activity Search (by Paul)



Front-End

In UserHomeView.vue:

```
searchActivity()  
searchByDate()  
sortActivity()
```



Back-End

In homeServer.js:

```
app.post("/searchActivity", function(req, res))  
app.post("/searchByDate", function(req, res))  
app.post("/MostRecent", function(req, res))
```

- *Admin Account (by Philip)*



Front-End

In AdminConsoleView.vue:

loadUserData()

In AdminAccountListView.vue:

loadAdminData()



Back-End

In adminServer.js:

deleteUser()

deleteAdmin()

createNewAdminAccount()