LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING

(AUTONOMOUS)

Estd. 1998

Approved by AICTE, New Delhi and Permanently affiliated to JNTUK, Kakinada

L.B. Reddy Nagar, Mylavaram, N.T.R. District, Andhra Pradesh-521230

ASE, CE, CSE, ECE, EEE, IT & ME Under Tier-I

NAAC A

CGPA: 3.20/4

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

15-DAY MERN STACK HANDS-ON ONLINE TRAINING PROGRAM

(Full-Stack Web Development | FSD-II)



PROGRAM DETAILS

Academic Year	: 2026 – 27
Programme	: 15-Day MERN Stack Hands-On Online Training Program
Dates	: 05 October 2026 – 30 October 2026 (15 sessions)
Time	: 06:30 PM – 07:30 PM
Mode / Platform	: Online – Microsoft Teams https://teams.microsoft.com/meet/498820073162907?p=Oe44JI4jt5gAnHQR9o
Target Participants	: CSE V Semester Students
Duration	: 15 Days | 15 Hours | Live Hands-on Sessions
Organised by	: Department of CSE, LBRCE, in association with Skelo AI
Convener	: Dr. S. Nagarjuna Reddy, Professor & Head, Department of CSE
Co-Conveners	: Dr. K. Devipriya, Professor; Mr. N. Srinivasa Rao, Sr. Assistant Professor
Coordinators	: Mr. Md. Amanatulla, Sr. Assistant Professor; Mrs. K. Naga Rajeswari, Assistant Professor
Trainer	: Vikram – Software Engineer, Full Stack Developer, Backend Developer and Technical Trainer

1. OBJECTIVE OF THE PROGRAMME

The primary objective of the 15-Day MERN Stack Hands-On Online Training Program was to provide CSE V Semester students with practical exposure to full-stack web application development using MongoDB, Express.js, React.js and Node.js. The programme was designed to strengthen FSD-II laboratory learning through live demonstrations, guided coding activities and progressive development of a full-stack application. Sessions were conducted live through Microsoft Teams from 06:30 PM to 07:30 PM, with each session contributing to a Student Task Manager capstone application.

2. LEARNING OBJECTIVES

- Understand Node.js and develop server-side applications using JavaScript.
- Design and implement RESTful APIs using Express.js.
- Create and manage cloud database collections using MongoDB Atlas and Mongoose.
- Develop component-based and interactive user interfaces using React.js.
- Integrate frontend, backend and database components into a full-stack application.
- Implement authentication and protected routes using bcrypt and JWT.
- Relate MERN concepts to prior learning in Java, Servlets, JSP and database connectivity.

3. PROGRAMME SCOPE AND TECHNOLOGY STACK

- Node.js – server-side JavaScript and backend project setup.
- Express.js – routing, middleware and REST API development.
- MongoDB with Mongoose – database design, schemas, models and CRUD operations.
- React.js – components, state, events, forms and routing.
- Integration and Capstone – JWT authentication, protected routes, testing and deployment overview.

4. DAY-WISE DESCRIPTION OF THE PROGRAMME

The programme was delivered over 15 one-hour live online sessions on the dates listed below. The day-wise topics and practical outcomes were aligned with the session-wise curriculum.

Day 1: Monday, 05-10-2026 – Introduction to Node.js and JavaScript Runtime

- Introduction to Node.js and server-side JavaScript.
- JavaScript outside the browser; V8 engine and event loop.
- Node.js compared with Java/Servlet request handling; REPL, CommonJS and ES modules.

Hands-on outcome: CLI script to read and write a JSON file using the fs module.

Day 2: Tuesday, 06-10-2026 – Asynchronous JavaScript and HTTP Server

- Callbacks, Promises and async/await.
- Node.js http module; request and response objects.
- Creating routes and returning JSON responses.

Hands-on outcome: Raw HTTP server returning JSON on multiple routes.

Day 3: Wednesday, 07-10-2026 – Node.js Project Setup and Backend Scaffold

- npm and package.json; npm scripts and nodemon.
- Environment variables using dotenv.
- Backend project folder structure.

Hands-on outcome: Capstone backend scaffolded and running.

Day 4: Thursday, 08-10-2026 – Express.js Setup and Routing

- Express application setup and routing (GET, POST, PUT, DELETE).
- Route/query parameters and JSON body parsing.
- Express routing compared with Servlet mapping.

Hands-on outcome: /api/tasks GET and POST endpoints using in-memory data.

Day 5: Friday, 09-10-2026 – Express Middleware and CRUD APIs

- Built-in, custom and third-party middleware.
- express.Router and MVC folder structure (routes/controllers).
- Task CRUD operations and API testing using Postman.

Hands-on outcome: Full CRUD API for tasks, tested using Postman.

Day 6: Monday, 12-10-2026 – REST API Design, Validation and Error Handling

- REST API design and HTTP status codes.
- Input validation and centralised error handling.
- Cross-Origin Resource Sharing (CORS).

Hands-on outcome: Validated and error-handled API ready for database integration.

Day 7: Tuesday, 13-10-2026 – MongoDB Atlas and NoSQL Fundamentals

- NoSQL vs SQL, documents and collections.
- MongoDB Atlas setup and MongoDB Compass.
- Basic database CRUD operations.

Hands-on outcome: Cloud database created; sample tasks inserted and queried.

Day 8: Wednesday, 14-10-2026 – Mongoose Schemas, Models and CRUD

- Connecting the application to MongoDB.
- Mongoose schemas, models, data types and validation.
- CRUD operations using Mongoose.

Hands-on outcome: Capstone API switched from in-memory data to MongoDB.

Day 9: Thursday, 15-10-2026 – MongoDB Queries and Relationships

- Query operators, filtering, sorting and pagination.
- Relationships using references and populate.
- User model and task filtering by status and priority.

Hands-on outcome: User model added; tasks linked to users and filtered by status and priority.

Day 10: Friday, 16-10-2026 – React.js Fundamentals and UI Components

- Single-page applications and React vs JSP rendering.
- Vite setup, JSX, components and props.
- Building Task List and Task Card interfaces.

Hands-on outcome: Static Task List and Task Card user interface.

Day 11: Monday, 26-10-2026 – React State, Events and Forms

- State using useState and event handling.
- Controlled forms, list rendering and keys.
- Conditional rendering.

Hands-on outcome: Add and delete tasks working in local state.

Day 12: Tuesday, 27-10-2026 – React API Integration

- useEffect and REST API calls using fetch/axios.
- Loading and error states.
- Connecting React frontend with Express API.

Hands-on outcome: React frontend reading and creating tasks through the Express API.

Day 13: Wednesday, 28-10-2026 – React Router and Full CRUD Integration

- React Router and multi-page navigation.
- Lifting state up.

- Task list, add task, edit and delete flows connected to backend.

Hands-on outcome: Task list, add task and edit task pages connected to the backend.

Day 14: Thursday, 29-10-2026 – Authentication and Protected Routes

- Password hashing with bcrypt.
- JWT registration and login.
- Protected API routes and protected React routes.

Hands-on outcome: Secure login; each user sees only their own tasks.

Day 15: Friday, 30-10-2026 – End-to-End Integration and Capstone Demonstration

- End-to-end application testing.
- Deployment overview (Render backend and Vercel frontend).
- Best practices, next steps and Student Task Manager capstone demonstration.

Hands-on outcome: Completed capstone demonstration and evaluation.

5. TRAINER PROFILE

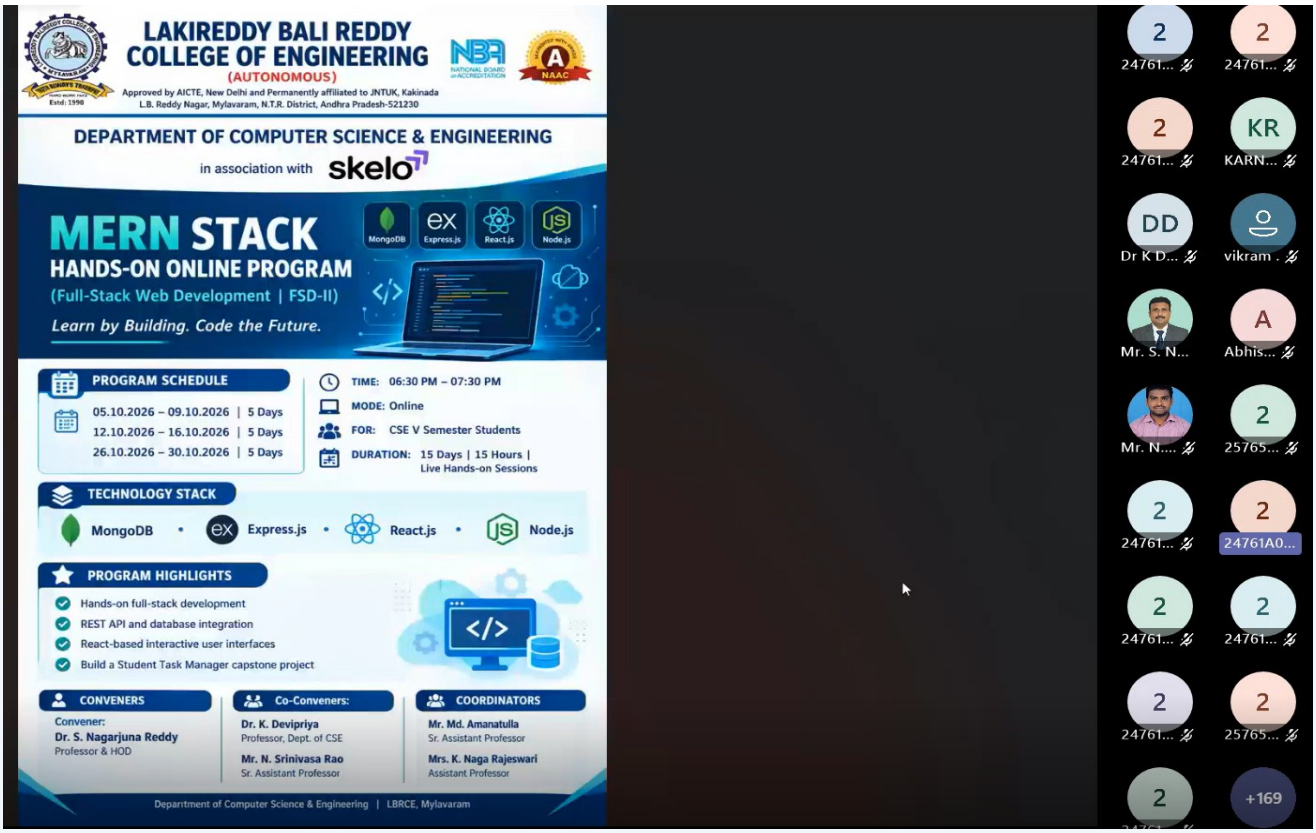
Vikram is a Software Engineer, Full Stack Developer, Backend Developer and Technical Trainer based in Anantapur, Andhra Pradesh. His profile reports training exposure to more than 5,000 students across engineering institutions. His technical skills include JavaScript, TypeScript, Python and Java; React.js, Node.js, Express.js, Spring Boot and FastAPI; MongoDB and PostgreSQL; and tools such as Git, Docker and Postman. His training experience includes MERN Stack corporate training for Tech Mahindra and software development training at multiple universities and institutions. The profile also records mentoring experience in real-world projects, code reviews and placement preparation.

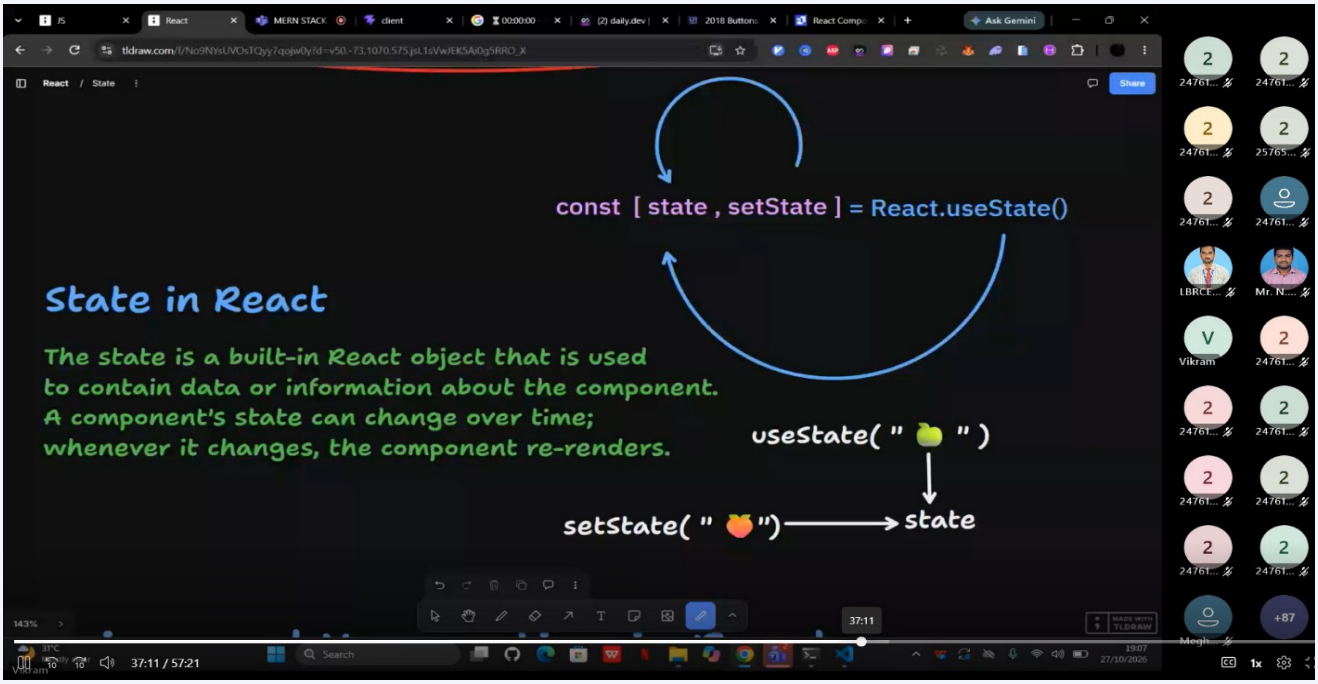
6. SKelo AI – FOUNDER PROFILE

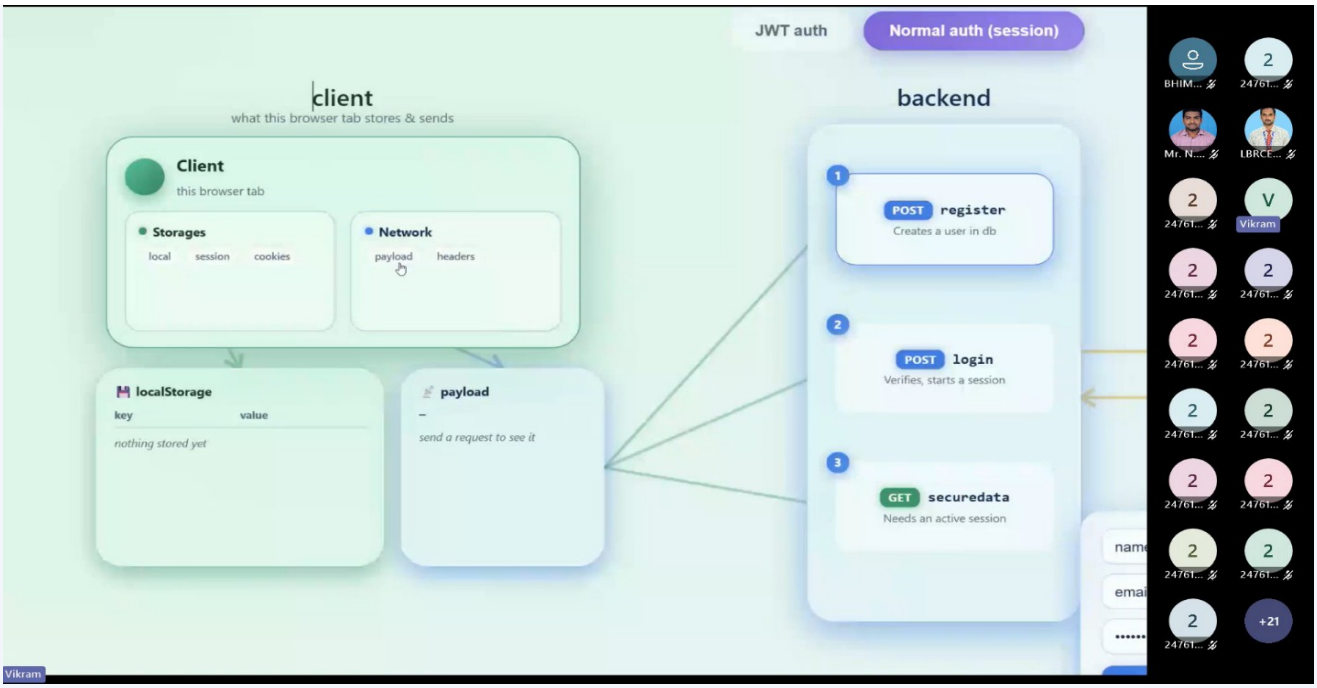
Abhishek Basavaraj Gowda is the Founder of Skelo AI, an AI-powered platform focused on transforming college training and placement systems through student engagement, real-time analytics and training tools. He is also the Co-Founder and CEO of Leantech Labs, where he works with businesses on growth strategies and cybersecurity solutions. His previous experience includes working as a Senior Software Development Engineer at Capullo School of Fine Arts & Designs. He holds a Bachelor's degree in Computer Science from JJDC.

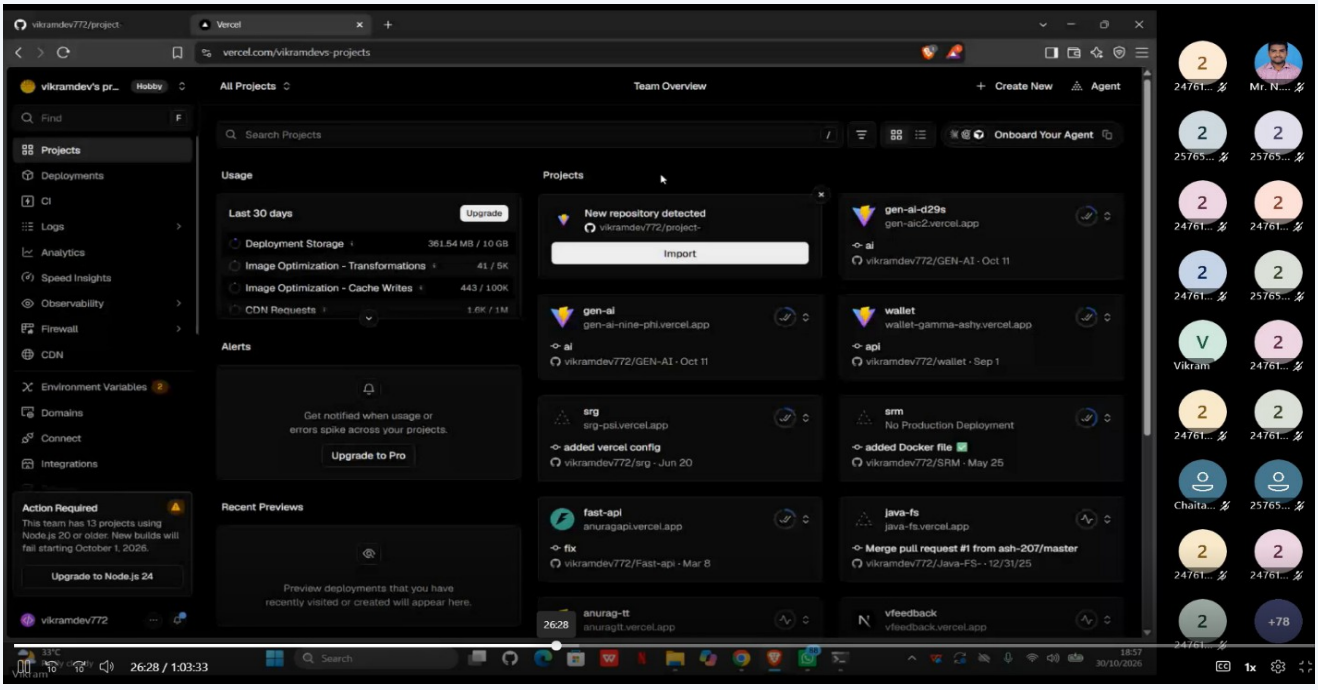
7. PHOTO DOCUMENTATION

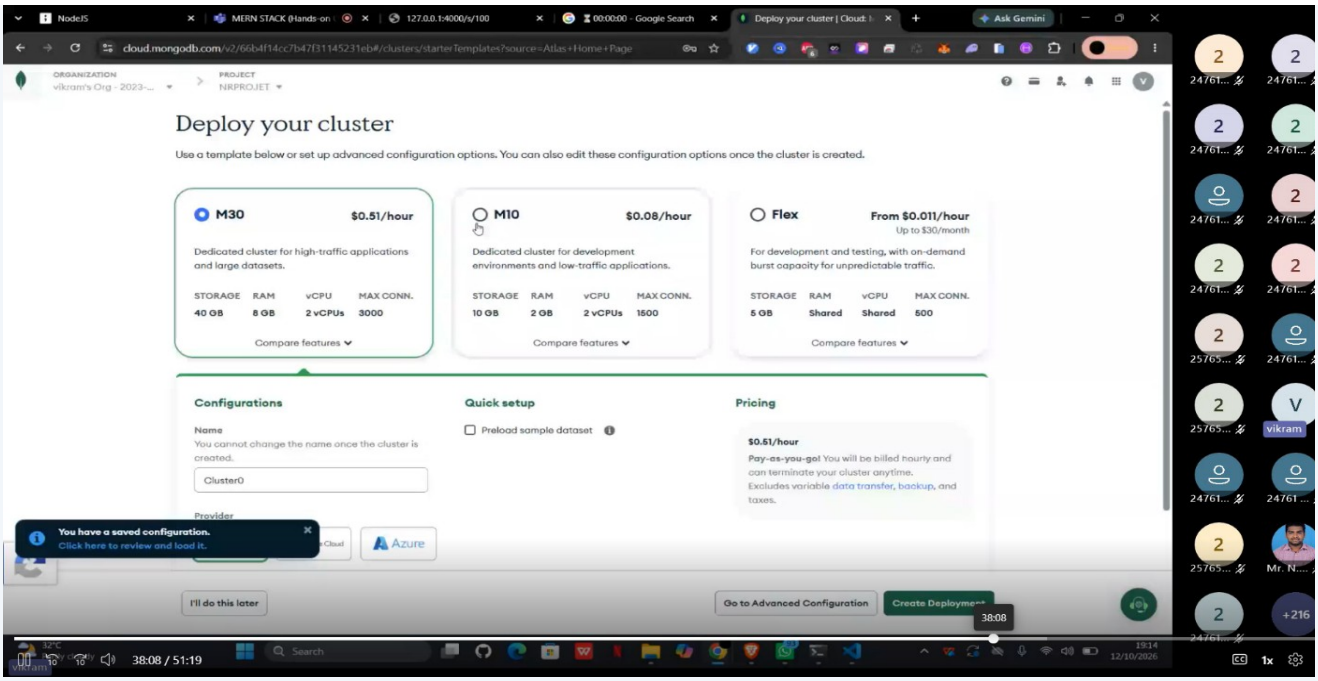
Photographs/screenshots of live Microsoft Teams sessions, trainer interactions, student participation, coding demonstrations and capstone activities may be inserted in the spaces provided under each day-wise report. Additional photographs may be added below.

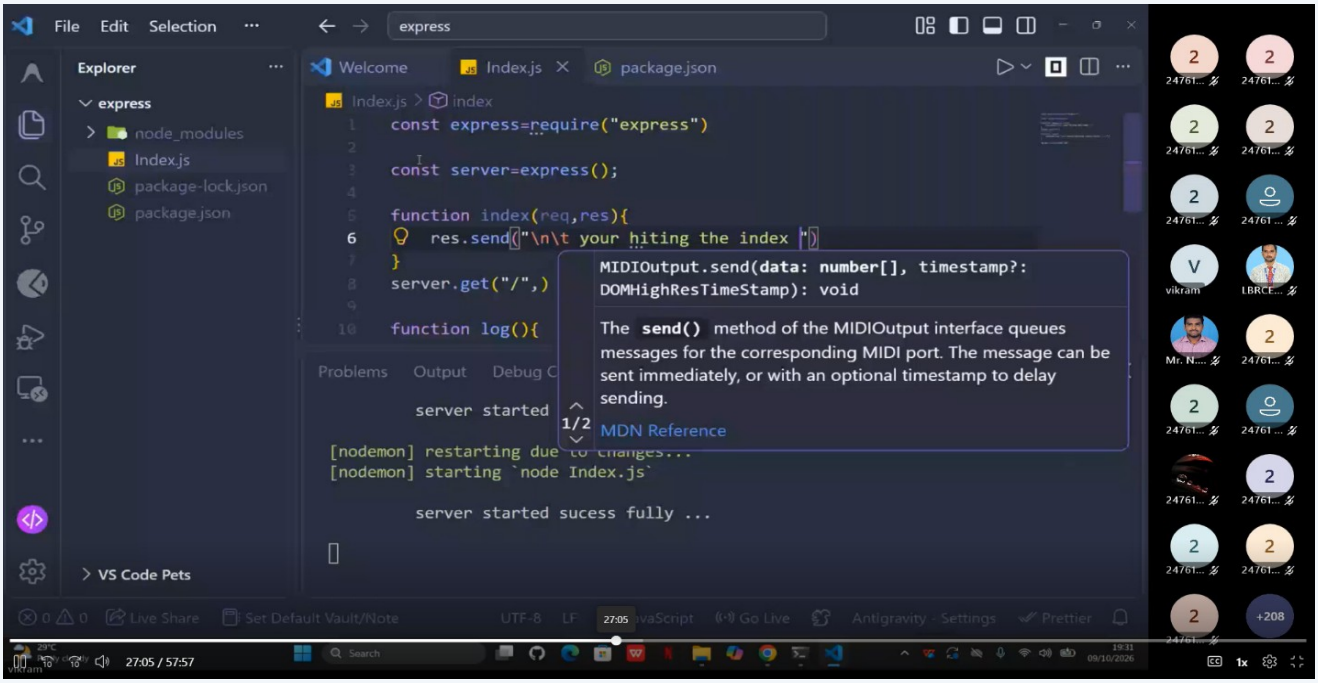

LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING
(AUTONOMOUS)
Approved by AICTE, New Delhi and Permanently affiliated to JNTUK, Kakinada
L.B. Reddy Nagar, Mylavaram, N.T.R. District, Andhra Pradesh-521230
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
in association with skelo
MERN STACK
HANDS-ON ONLINE PROGRAM
(Full-Stack Web Development | FSD-II)
Learn by Building. Code the Future.
MongoDB
Express.js
React.js
Node.js
PROGRAM SCHEDULE
05.10.2026 – 09.10.2026 | 5 Days
12.10.2026 – 16.10.2026 | 5 Days
26.10.2026 – 30.10.2026 | 5 Days
TIME: 06:30 PM – 07:30 PM
MODE: Online
FOR: CSE V Semester Students
DURATION: 15 Days | 15 Hours | Live Hands-on Sessions
TECHNOLOGY STACK
MongoDB • Express.js • React.js • Node.js
PROGRAM HIGHLIGHTS
Hands-on full-stack development
REST API and database integration
React-based interactive user interfaces
Build a Student Task Manager capstone project
CONVENERS
Convener:
Dr. S. Nagarjuna Reddy
Professor & HOD
Co-Conveners:
Dr. K. Devipriya
Professor, Dept. of CSE
Mr. N. Srinivasa Rao
Sr. Assistant Professor
COORDINATORS
Mr. Md. Amanatulla
Sr. Assistant Professor
Mrs. K. Naga Rajeswari
Assistant Professor
Department of Computer Science & Engineering | LBRCE, Mylavaram

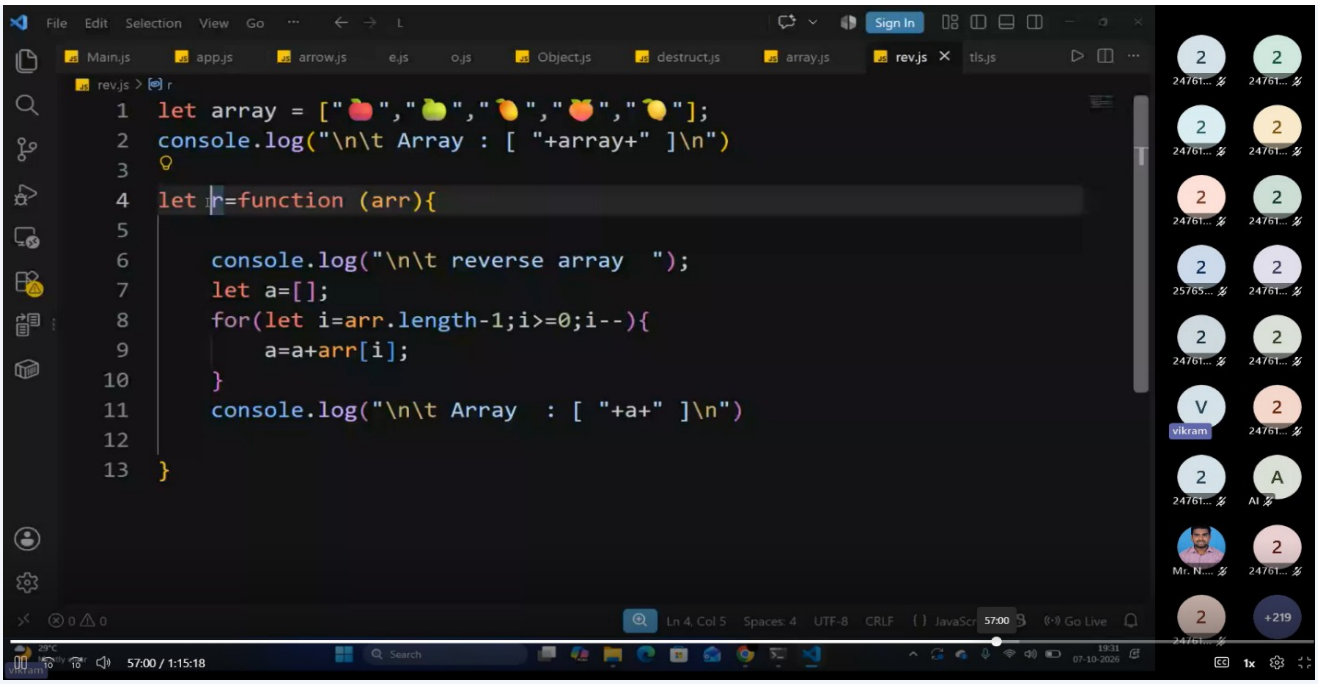

const [state , setState] = React.useState()
State in React
The state is a built-in React object that is used
to contain data or information about the component.
A component's state can change over time;
whenever it changes, the component re-renders.
useState(" ")
setState(" ")
state
37:11 / 57:21

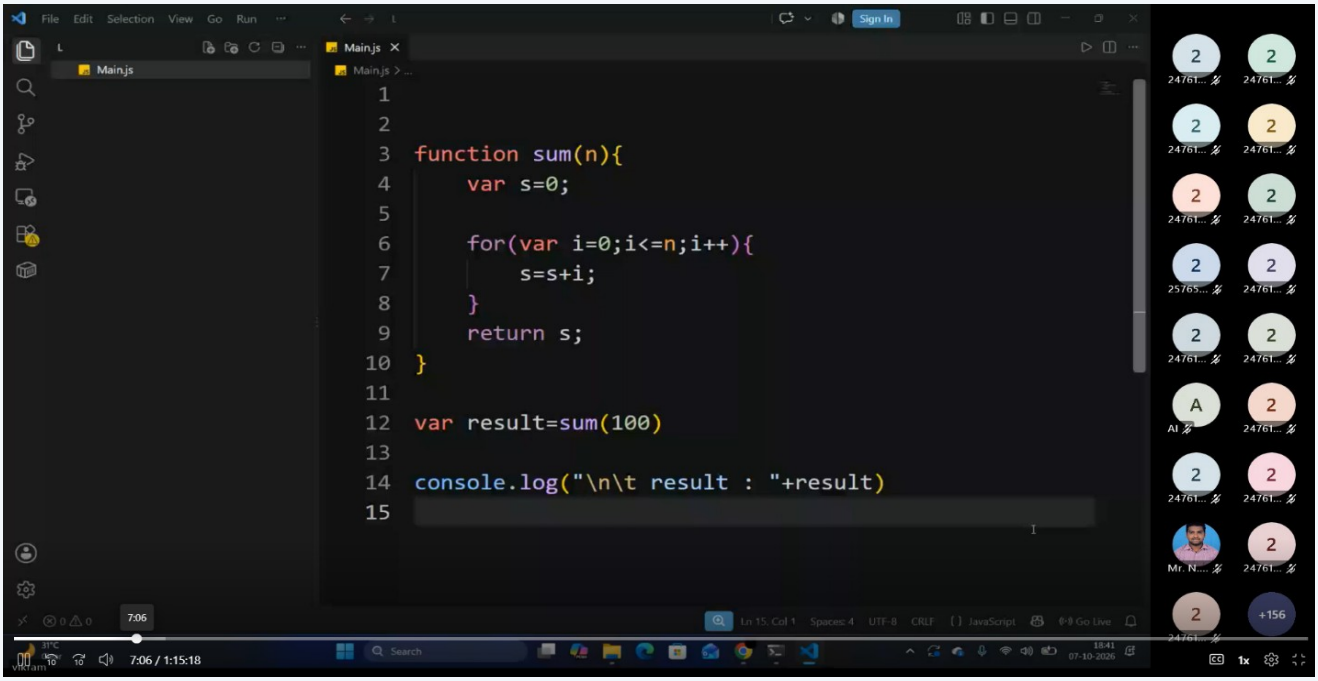

JWT auth
Normal auth (session)
client
what this browser tab stores & sends
backend
Client
this browser tab
Storages
local
session
cookies
Network
payload
headers
localStorage
key
value
nothing stored yet
payload
–
send a request to see it
1
POST register
Creates a user in db
2
POST login
Verifies, starts a session
3
GET securedata
Needs an active session
Vikram

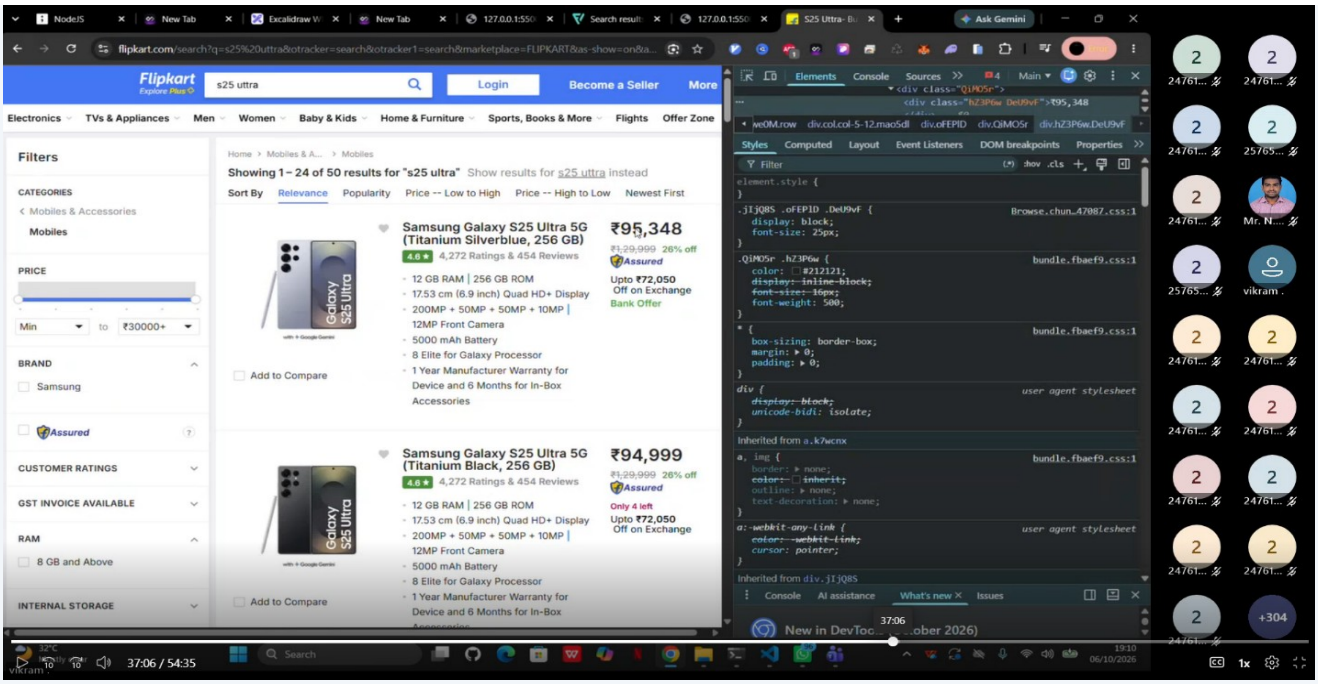

vercel.com/vikramdevs-projects
All Projects
Team Overview
Create New
Agent
Search Projects
Onboard Your Agent
Projects
Deployments
CI
Logs
Analytics
Speed Insights
Observability
Firewall
CDN
Environment Variables
Domains
Connect
Integrations
Usage
Last 30 days
Upgrade
Deployment Storage
361.54 MB / 10 GB
Image Optimization - Transformations
41 / 5K
Image Optimization - Cache Writes
443 / 100K
CDN Requests
1.6K / 1M
Alerts
Get notified when usage or errors spike across your projects.
Upgrade to Pro
Action Required
This team has 13 projects using Node.js 20 or older. New builds will fail starting October 1, 2026.
Upgrade to Node.js 24
Recent Previews
Preview deployments that you have recently visited or created will appear here.
New repository detected
Import
gen-ai-d29s
gen-aic2.vercel.app
gen-ai
gen-ai-nine-phi.vercel.app
wallet
wallet-gamma-ashy.vercel.app
srg
srg-psi.vercel.app
added vercel config
srm
No Production Deployment
added Docker file
fast-api
anuragapi.vercel.app
java-fs
java-fs.vercel.app
Merge pull request #1 from ash-207/master
anurag-tt
vfeedback
26:28 / 1:03:33
Vikram

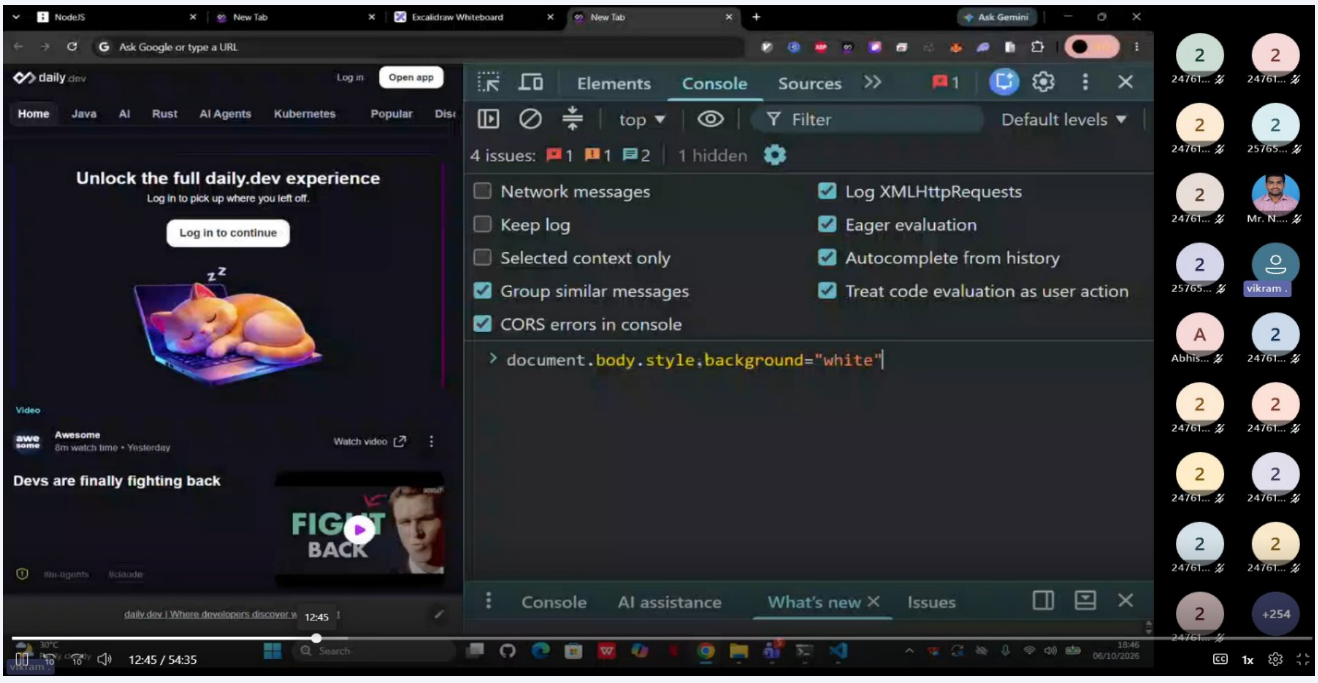

Deploy your cluster
Use a template below or set up advanced configuration options. You can also edit these configuration options once the cluster is created.
M30 $0.51/hour
Dedicated cluster for high-traffic applications and large datasets.
STORAGE RAM vCPU MAX CONN.
40 GB 8 GB 2 vCPUs 3000
Compare features
M10 $0.08/hour
Dedicated cluster for development environments and low-traffic applications.
STORAGE RAM vCPU MAX CONN.
10 GB 2 GB 2 vCPUs 1500
Compare features
Flex From $0.011/hour
Up to $30/month
For development and testing, with on-demand burst capacity for unpredictable traffic.
STORAGE RAM vCPU MAX CONN.
5 GB Shared Shared 500
Compare features
Configurations
Name
You cannot change the name once the cluster is created.
Cluster0
Provider
Quick setup
Preload sample dataset
Pricing
$0.51/hour
Pay-as-you-go! You will be billed hourly and can terminate your cluster anytime.
Excludes variable data transfer, backup, and taxes.
You have a saved configuration.
Click here to review and load it.
Azure
I'll do this later
Go to Advanced Configuration
Create Deployment
38:08
38:08 / 51:19

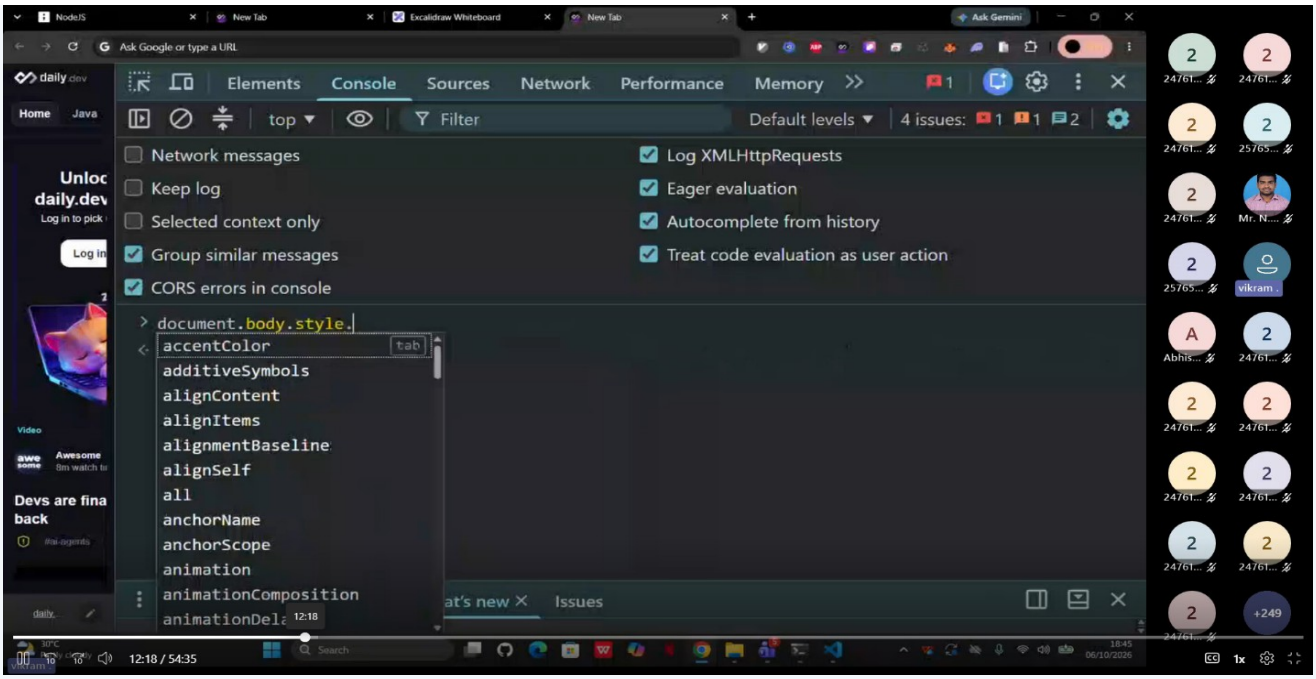

File Edit Selection
express
Explorer
express
node_modules
Index.js
package-lock.json
package.json
Welcome
Index.js
package.json
const express=require("express")
const server=express();
function index(req,res){
res.send("\n\t your hiting the index ")
}
server.get("/",)
function log(){
MIDIOutput.send(data: number[], timestamp?: DOMHighResTimeStamp): void
The send() method of the MIDIOutput interface queues messages for the corresponding MIDI port. The message can be sent immediately, or with an optional timestamp to delay sending.
MDN Reference
1/2
Problems Output
server started
[nodemon] restarting due to changes...
[nodemon] starting `node Index.js`
server started sucess fully ...
VS Code Pets
27:05
27:05 / 57:57


File Edit Selection View Go
Sign In
Main.js app.js arrow.js e.js o.js Object.js destruct.js array.js rev.js tls.js
rev.js > r
let array = ["🍅","🍏","🍊","🍑","🍋"];
console.log("\n\t Array : ["+array+"]\n")
let r=function (arr){
console.log("\n\t reverse array ");
let a=[];
for(let i=arr.length-1;i>=0;i--){
a=a+arr[i];
}
console.log("\n\t Array : ["+a+"]\n")
}
Ln 4, Col 5 Spaces: 4 UTF-8 CRLF JavaScr 57:00 Go Live
57:00 / 1:15:18
vikram
+219

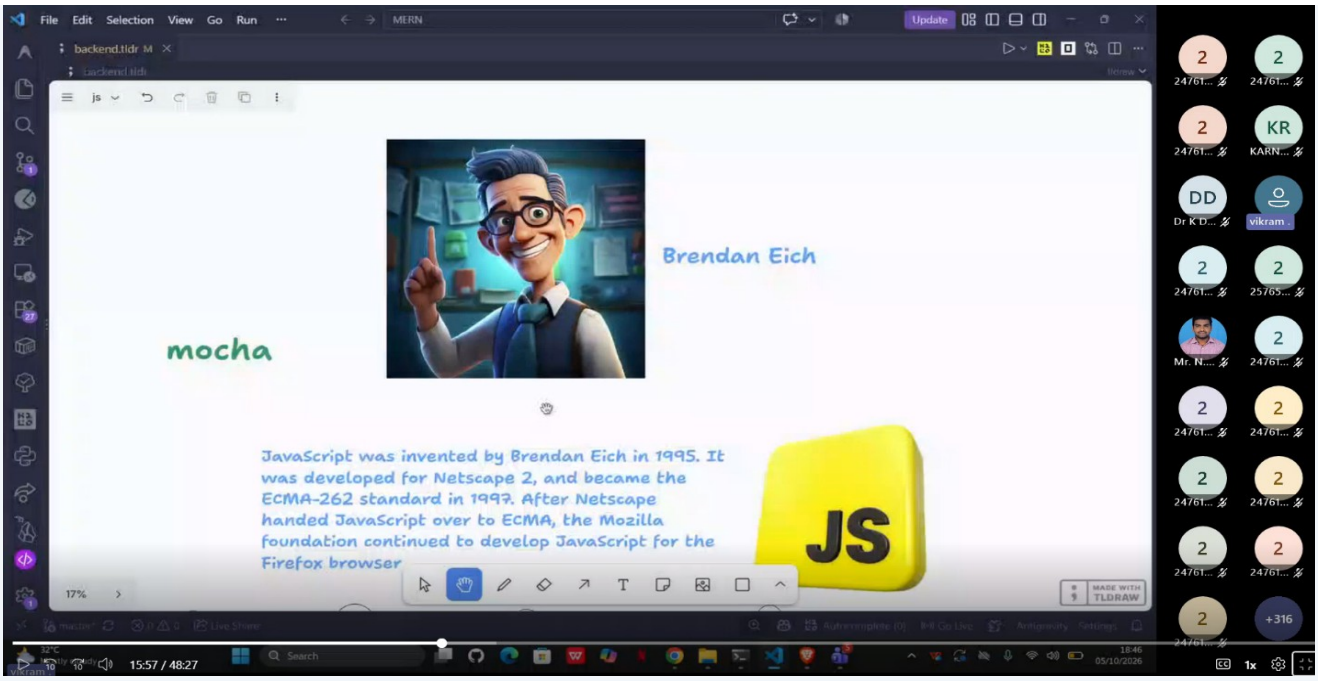

File Edit Selection View Go Run
Sign In
Main.js
Main.js
function sum(n){
var s=0;
for(var i=0;i<=n;i++){
s=s+i;
}
return s;
}
var result=sum(100)
console.log("\n\t result : "+result)
Ln 15, Col 1 Spaces: 4 UTF-8 CRLF JavaScript Go Live
7:06
7:06 / 1:15:18
vikram
+156

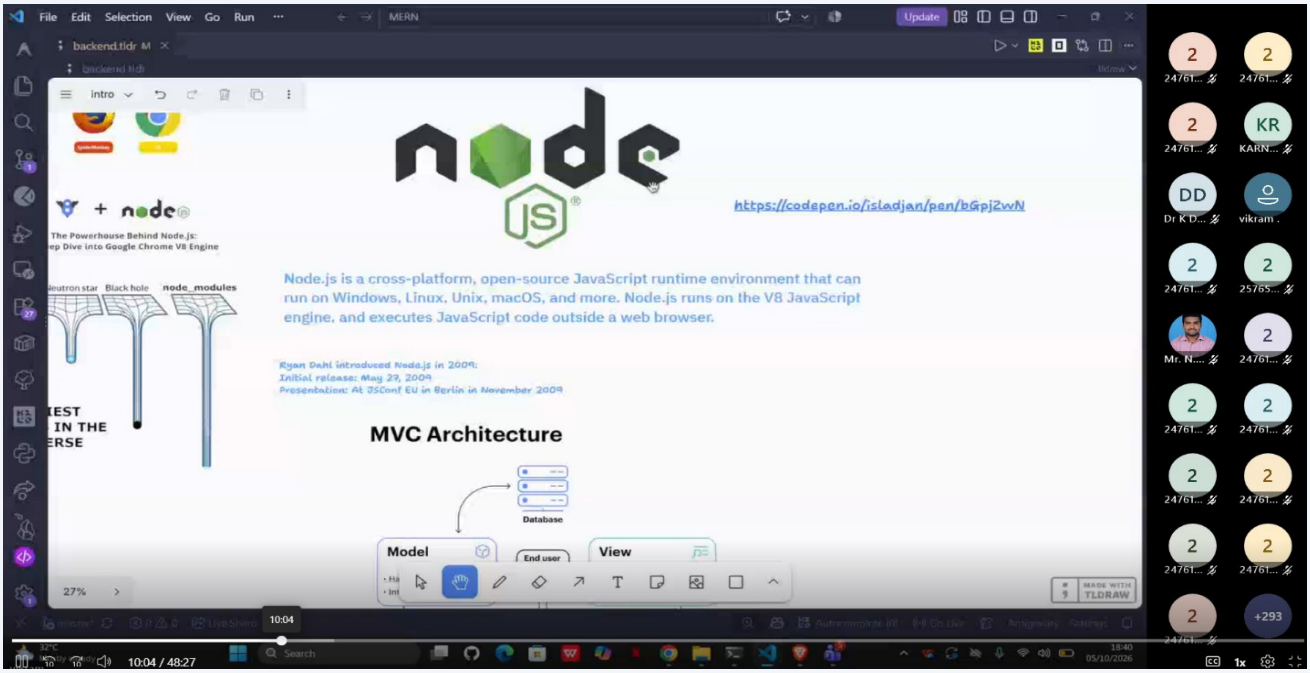

Flipkart
s25 uttra
Login
Become a Seller
More
Electronics
TVs & Appliances
Men
Women
Baby & Kids
Home & Furniture
Sports, Books & More
Flights
Offer Zone
Filters
CATEGORIES
Mobiles & Accessories
Mobiles
PRICE
BRAND
Samsung
CUSTOMER RATINGS
GST INVOICE AVAILABLE
RAM
8 GB and Above
INTERNAL STORAGE
Showing 1 – 24 of 50 results for "s25 ultra"
Sort By
Relevance
Popularity
Price -- Low to High
Price -- High to Low
Newest First
Samsung Galaxy S25 Ultra 5G (Titanium Silverblue, 256 GB)
4.6★ 4,272 Ratings & 454 Reviews
12 GB RAM | 256 GB ROM
17.53 cm (6.9 inch) Quad HD+ Display
200MP + 50MP + 50MP + 10MP | 12MP Front Camera
5000 mAh Battery
8 Elite for Galaxy Processor
1 Year Manufacturer Warranty for Device and 6 Months for In-Box Accessories
₹95,348
₹1,29,999 26% off
Assured
Upto ₹72,050 Off on Exchange
Bank Offer
Add to Compare
Samsung Galaxy S25 Ultra 5G (Titanium Black, 256 GB)
4.6★ 4,272 Ratings & 454 Reviews
12 GB RAM | 256 GB ROM
17.53 cm (6.9 inch) Quad HD+ Display
200MP + 50MP + 50MP + 10MP | 12MP Front Camera
5000 mAh Battery
8 Elite for Galaxy Processor
1 Year Manufacturer Warranty for Device and 6 Months for In-Box Accessories
₹94,999
₹1,29,999 26% off
Assured
Only 4 left
Upto ₹72,050 Off on Exchange
Add to Compare
Elements
Console
Sources
Styles
Computed
Layout
Event Listeners
DOM breakpoints
Properties
Console
AI assistance
What's new
Issues
37:06 / 54:35

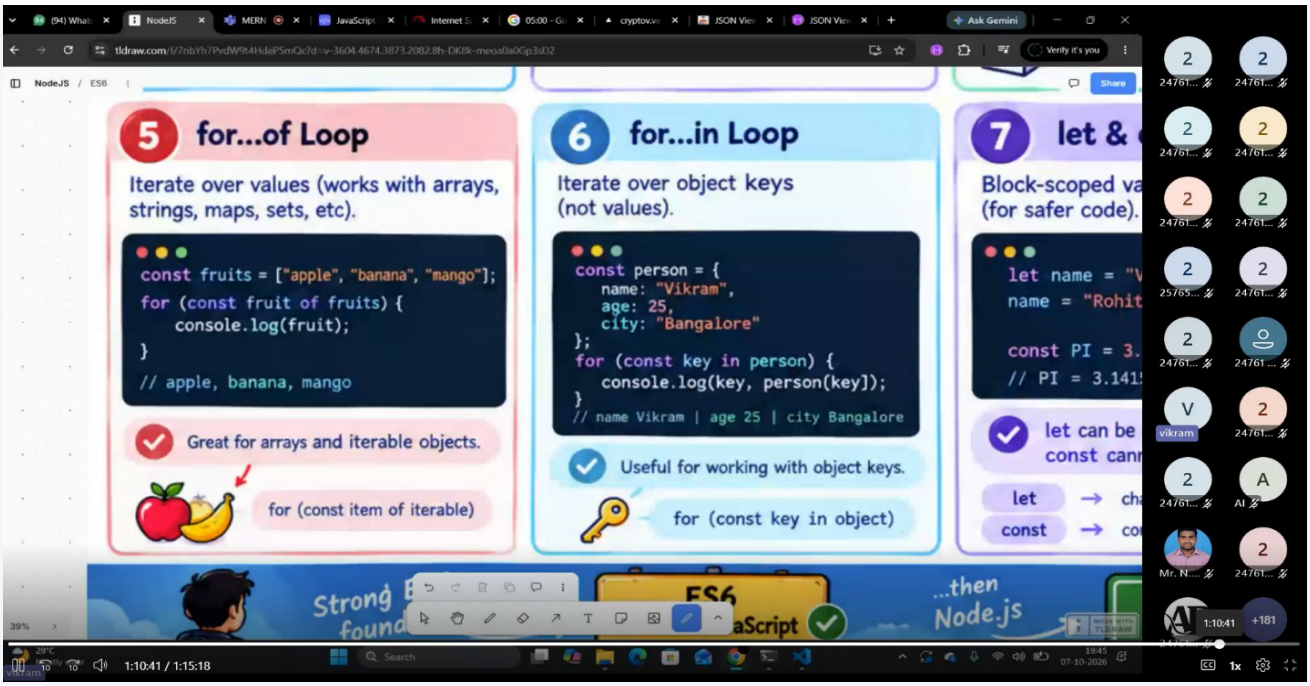

Ask Google or type a URL
daily.dev
Log in
Open app
Home
Java
AI
Rust
AI Agents
Kubernetes
Popular
Unlock the full daily.dev experience
Log in to pick up where you left off.
Log in to continue
Video
Awesome
8m watch time • Yesterday
Watch video
Devs are finally fighting back
FIGHT BACK
Elements
Console
Sources
top
Filter
Default levels
4 issues: 1 1 2 1 hidden
Network messages
Keep log
Selected context only
Group similar messages
CORS errors in console
Log XMLHttpRequests
Eager evaluation
Autocomplete from history
Treat code evaluation as user action
document.body.style.background="white"
Console
AI assistance
What's new
Issues
12:45 / 54:35


Elements
Console
Sources
Network
Performance
Memory
Network messages
Keep log
Selected context only
Group similar messages
CORS errors in console
Log XMLHttpRequests
Eager evaluation
Autocomplete from history
Treat code evaluation as user action
document.body.style.
accentColor
additiveSymbols
alignContent
alignItems
alignmentBaseline
alignSelf
all
anchorName
anchorScope
animation
animationComposition
Issues
12:18 / 54:35


JavaScript was invented by Brendan Eich in 1995. It was developed for Netscape 2, and became the ECMA-262 standard in 1997. After Netscape handed JavaScript over to ECMA, the Mozilla foundation continued to develop JavaScript for the Firefox browser
JS
European Computer Manufacturers Association
1997 1998 1999 2009 2015
ES1
ES2
ES3
ES5
ES6
ECMA Script
16:39 / 48:27


Brendan Eich
mocha
JavaScript was invented by Brendan Eich in 1995. It was developed for Netscape 2, and became the ECMA-262 standard in 1997. After Netscape handed JavaScript over to ECMA, the Mozilla foundation continued to develop JavaScript for the Firefox browser
JS
15:57 / 48:27


https://codepen.io/isladjan/pen/bGpjZwN
Node.js is a cross-platform, open-source JavaScript runtime environment that can run on Windows, Linux, Unix, macOS, and more. Node.js runs on the V8 JavaScript engine, and executes JavaScript code outside a web browser.
Ryan Dahl introduced Node.js in 2009:
Initial release: May 29, 2009
Presentation: At JSConf EU in Berlin in November 2009
MVC Architecture
Database
Model
End user
View
10:04 / 48:27


5 for...of Loop
Iterate over values (works with arrays, strings, maps, sets, etc).
const fruits = ["apple", "banana", "mango"];
for (const fruit of fruits) {
console.log(fruit);
}
// apple, banana, mango
Great for arrays and iterable objects.
for (const item of iterable)
6 for...in Loop
Iterate over object keys (not values).
const person = {
name: "Vikram",
age: 25,
city: "Bangalore"
};
for (const key in person) {
console.log(key, person(key));
}
// name Vikram | age 25 | city Bangalore
Useful for working with object keys.
for (const key in object)
7 let &
Block-scoped va
(for safer code).
let name = "V
name = "Rohit
const PI = 3.
// PI = 3.141
let can be
const can
let
const
Strong
found
ES6
aScript
...then
Node.js
1:10:41 / 1:15:18
1:10:41
+181
1x


24761A05E9
24761A05J1
25765A4207
KARNATI.NAGA RAJESWARI
Dr K DeviPriya
vikram .
Mr. S. Nagarjuna Reddy
Abhishek - Skelo AI
Mr. N. Srinivasa Rao
+174

8. IMPACT ANALYSIS

- Provided structured practical exposure to the four core technologies of the MERN stack.
- Introduced backend services and REST API development through guided coding activities.
- Provided exposure to cloud database management using MongoDB Atlas and Mongoose.
- Addressed component-based UI development, state management, forms and routing using React.js.
- Connected frontend, backend and database concepts through a progressive capstone workflow.
- Supported application-oriented learning relevant to the FSD-II laboratory.

9. FEEDBACK AND SUGGESTIONS

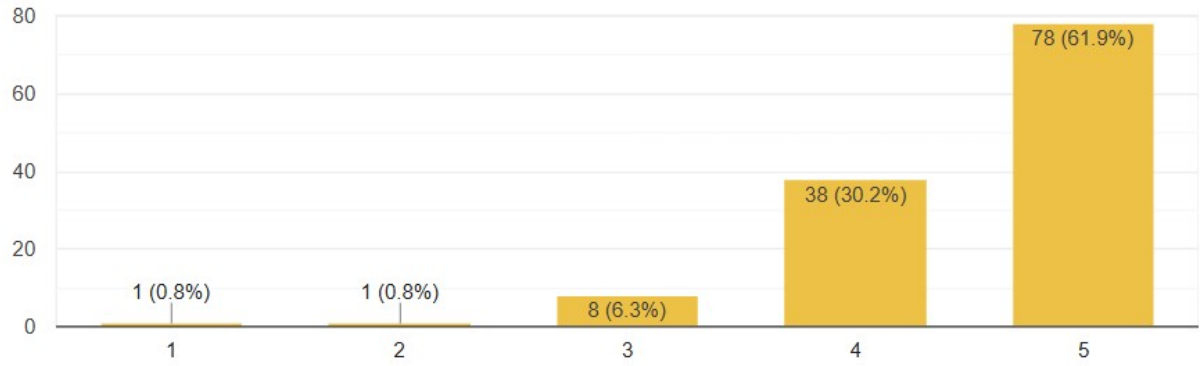
- Continue hands-on, project-oriented training aligned with the FSD laboratory curriculum.
- Provide additional practice time for API debugging, database integration and frontend-backend connectivity.
- Encourage students to complete and enhance the Student Task Manager application.
- Include further sessions on deployment, testing, security and industry practices.

Day 1: Monday, 05-10-2026 – Introduction to Node.js and JavaScript Runtime

How effectively did the trainer explain Node.js, server-side JavaScript and the V8 engine?

Copy chart

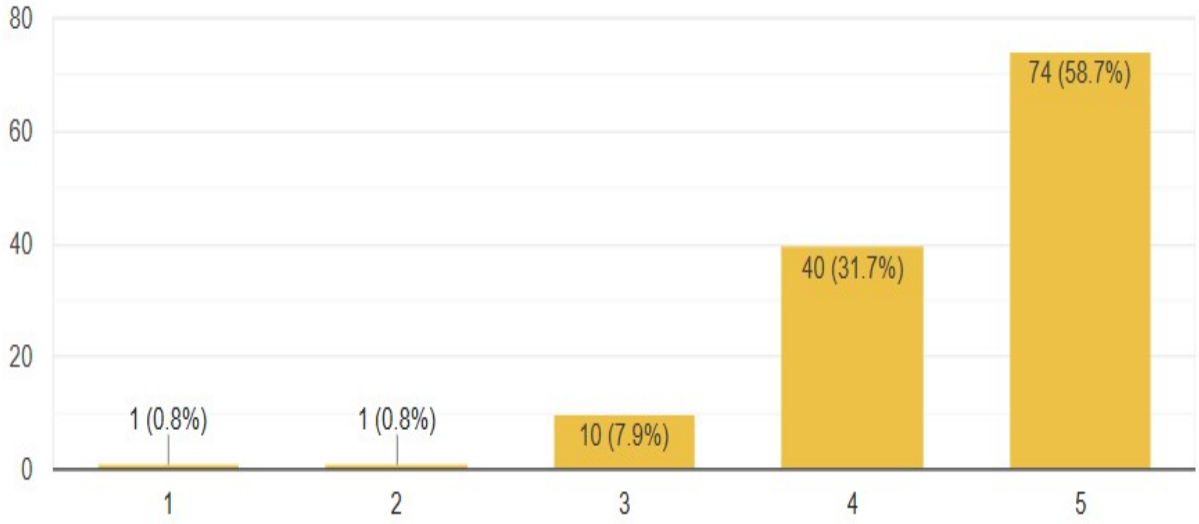
126 responses



Copy chart

How well did you understand the Event Loop, REPL, CommonJS and ES Modules?

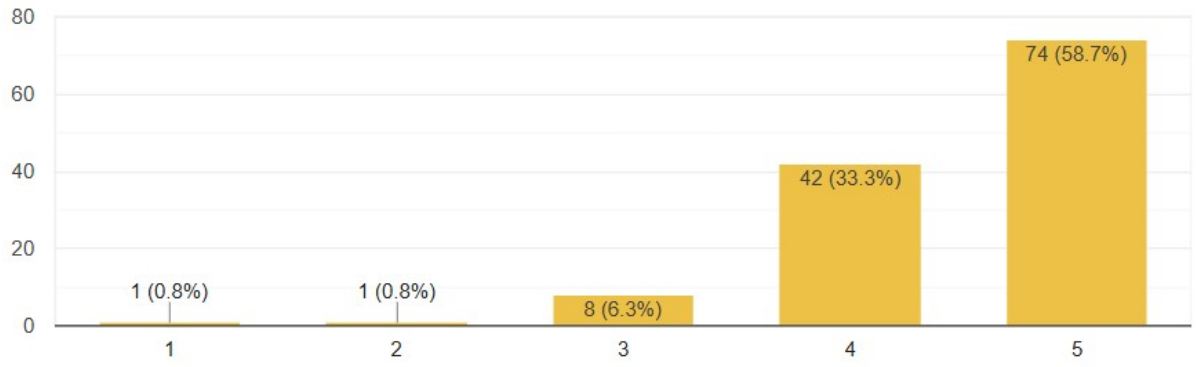
126 responses



How useful was the hands-on session on reading and writing JSON files using the fs module?

Copy chart

126 responses

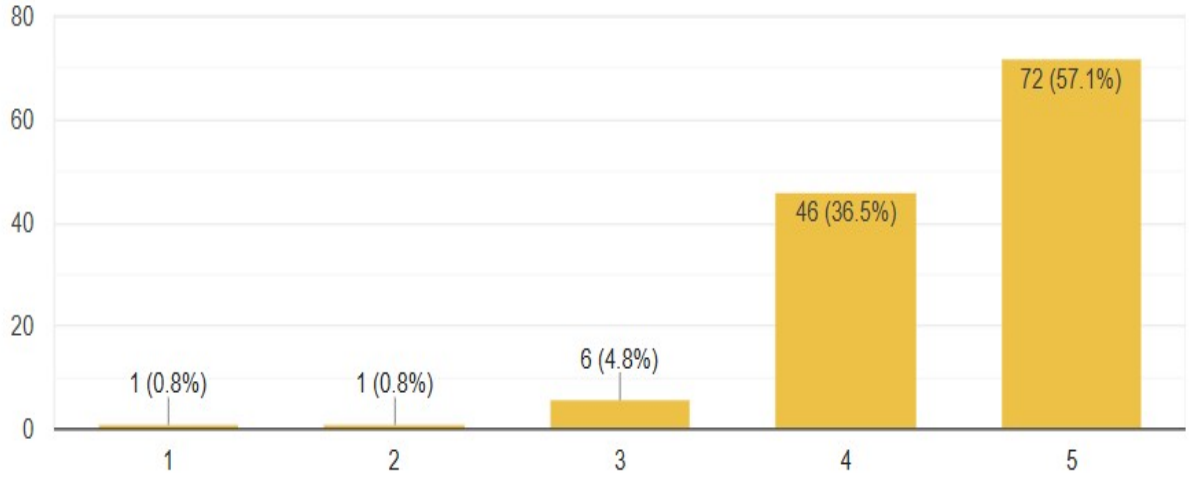


Day 2: Tuesday, 06-10-2026 – Asynchronous JavaScript and HTTP Server

How effectively were Callbacks, Promises and async/await concepts explained?

Copy chart

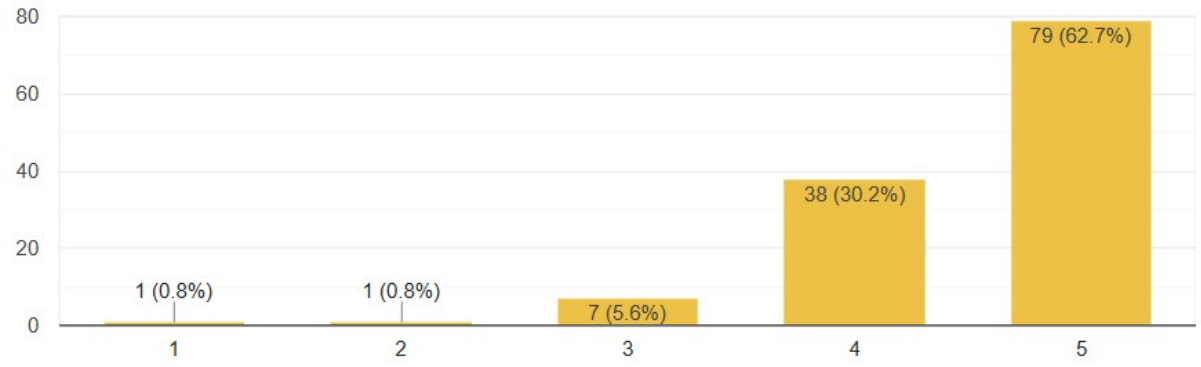
126 responses



Copy chart

How well did you understand the Node.js HTTP module and request-response handling?

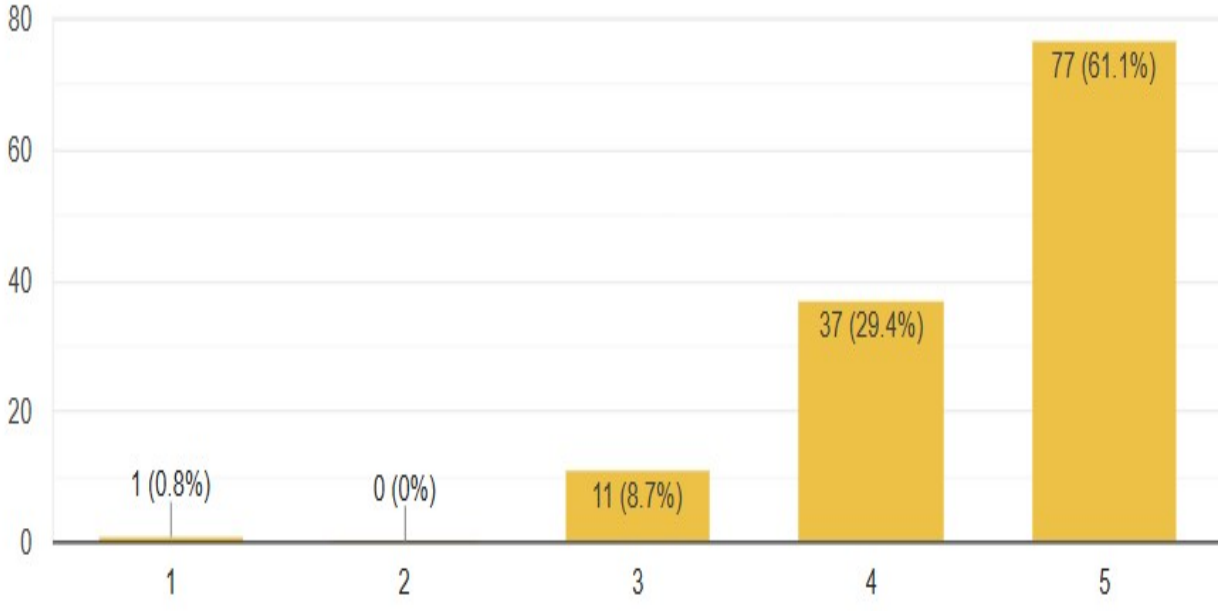
126 responses



How useful was the practical session on creating an HTTP server with multiple JSON routes?

Copy chart

126 responses

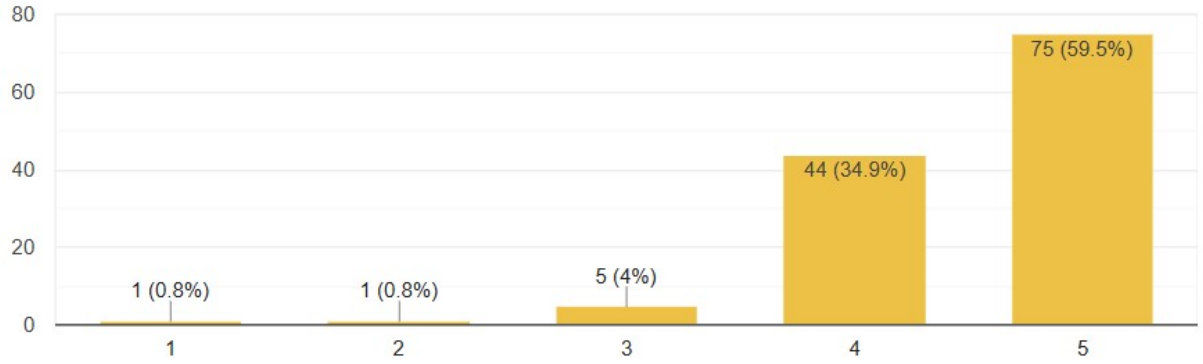


Day 3: Wednesday, 07-10-2026 – Node.js Project Setup and Backend Scaffold

How clearly were npm, package.json, npm scripts and nodemon explained?

Copy chart

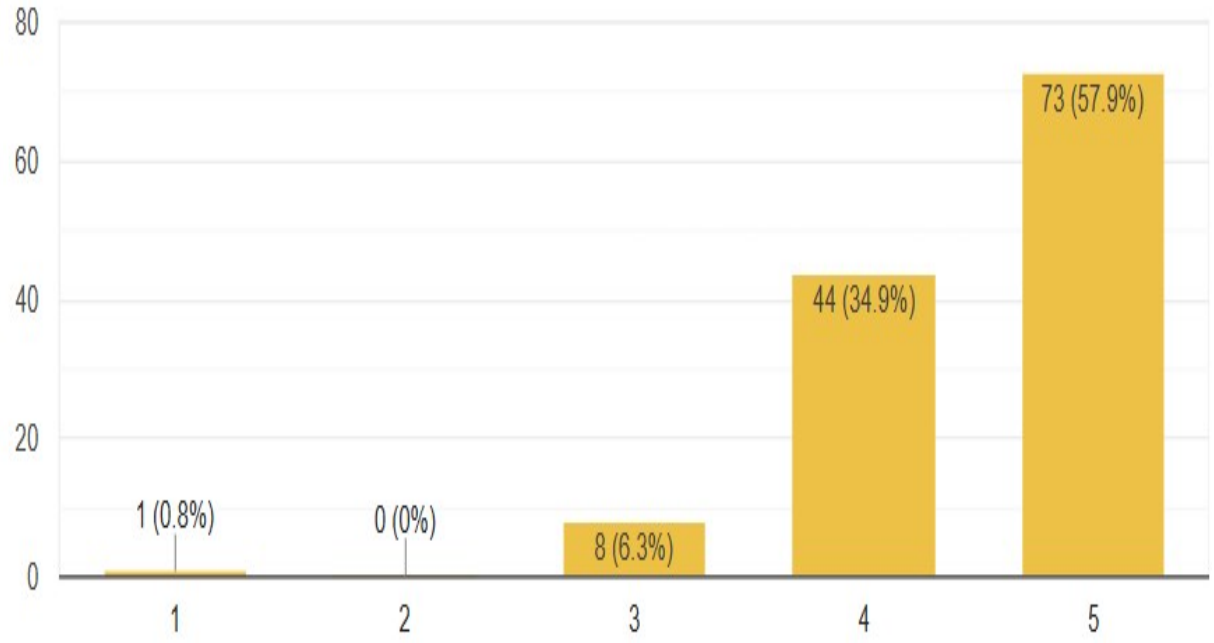
126 responses



How well did you understand environment variables and the dotenv package?

Copy chart

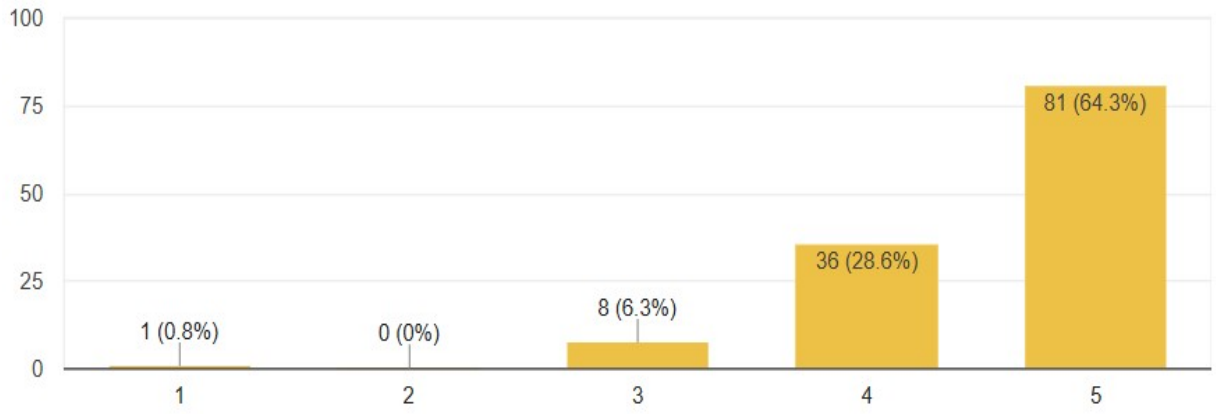
126 responses



How useful was the hands-on session on creating and running the backend project structure?

Copy chart

126 responses

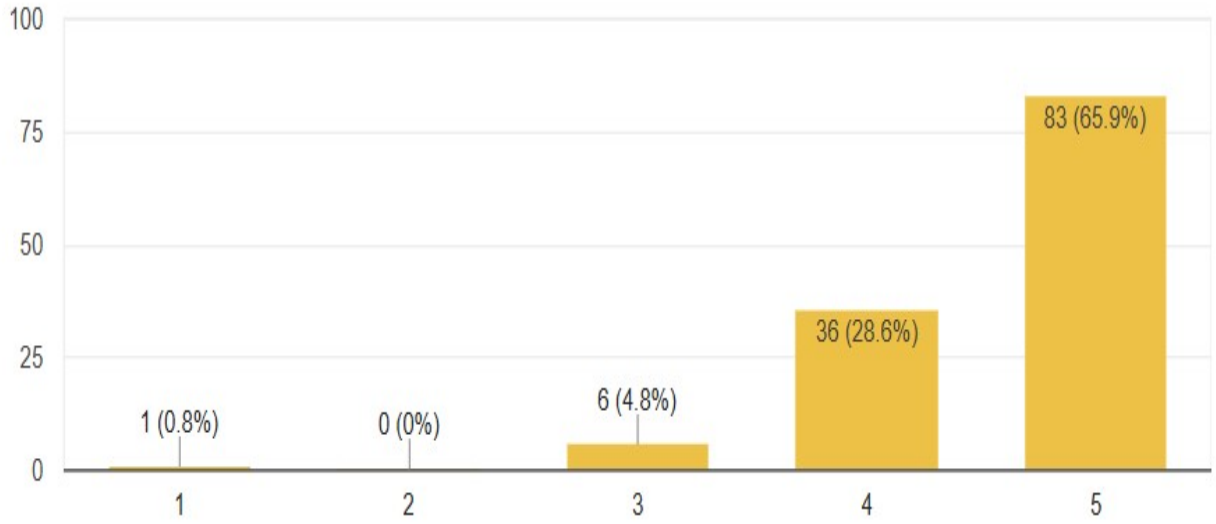


Day 4: Thursday, 08-10-2026 – Express.js Setup and Routing

How effectively did the trainer explain Express.js application setup and HTTP methods (GET, POST, PUT and DELETE)?

Copy chart

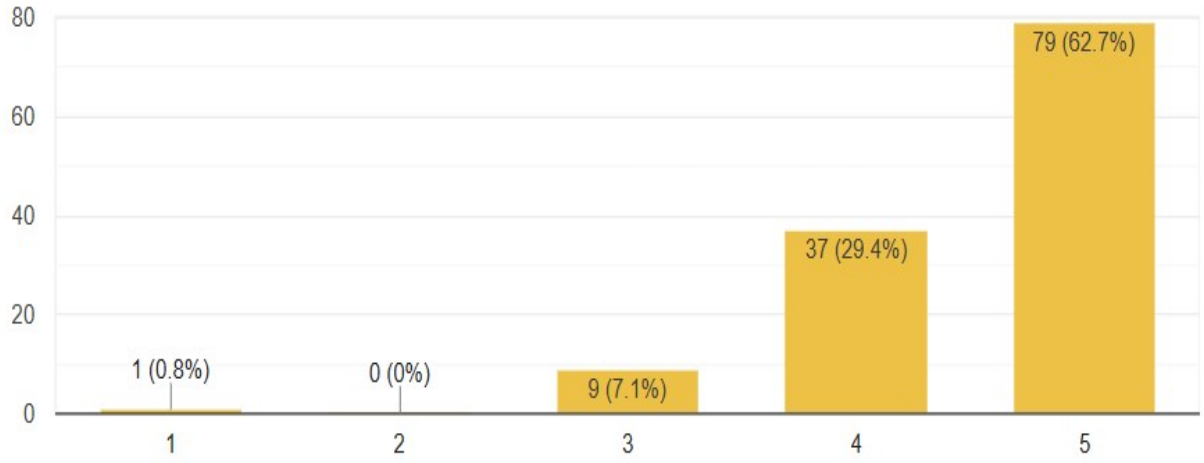
126 responses



How well did you understand route parameters, query parameters and JSON body parsing?

Copy chart

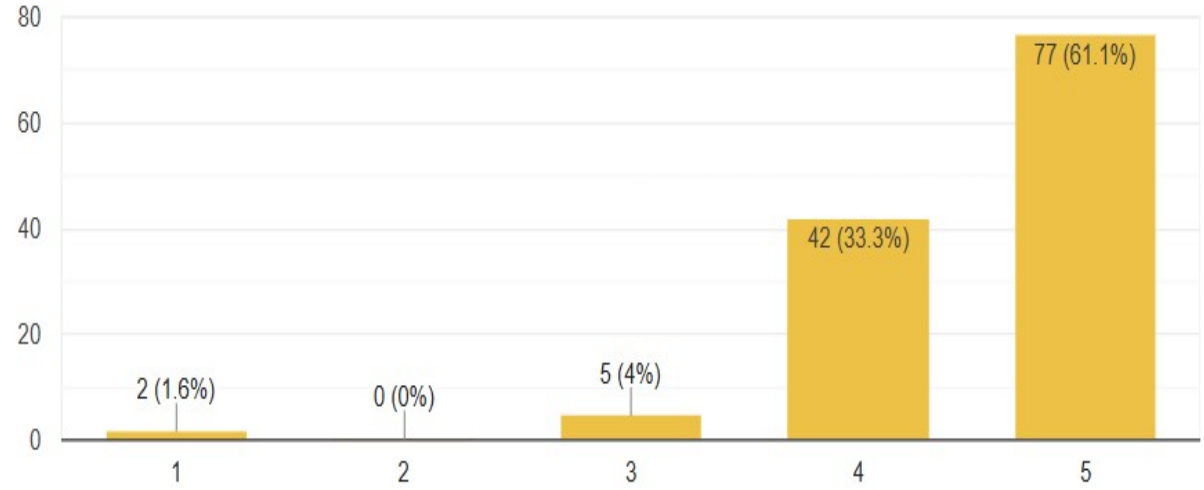
126 responses



How useful was the practical implementation of GET and POST API endpoints for tasks?

Copy chart

126 responses

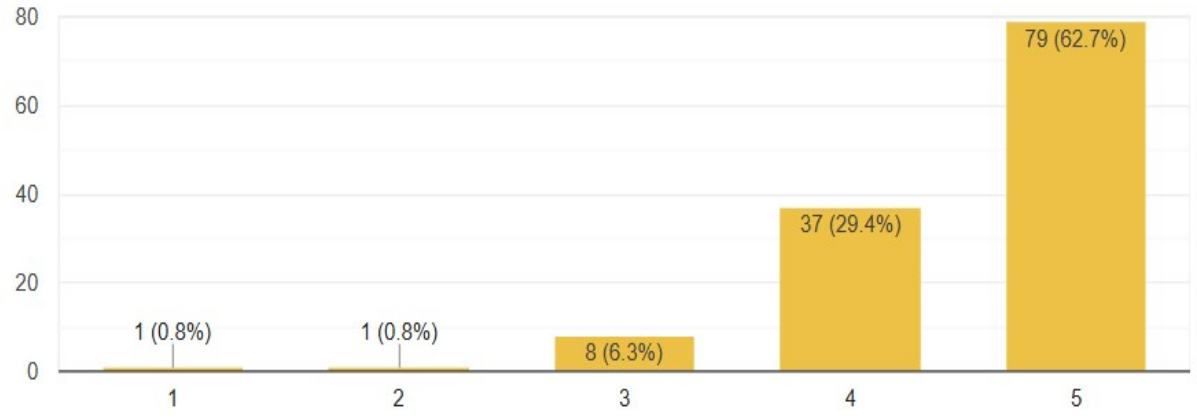


Day 5: Friday, 09-10-2026 – Express Middleware and CRUD APIs

How clearly were built-in, custom and third-party middleware concepts explained?

Copy chart

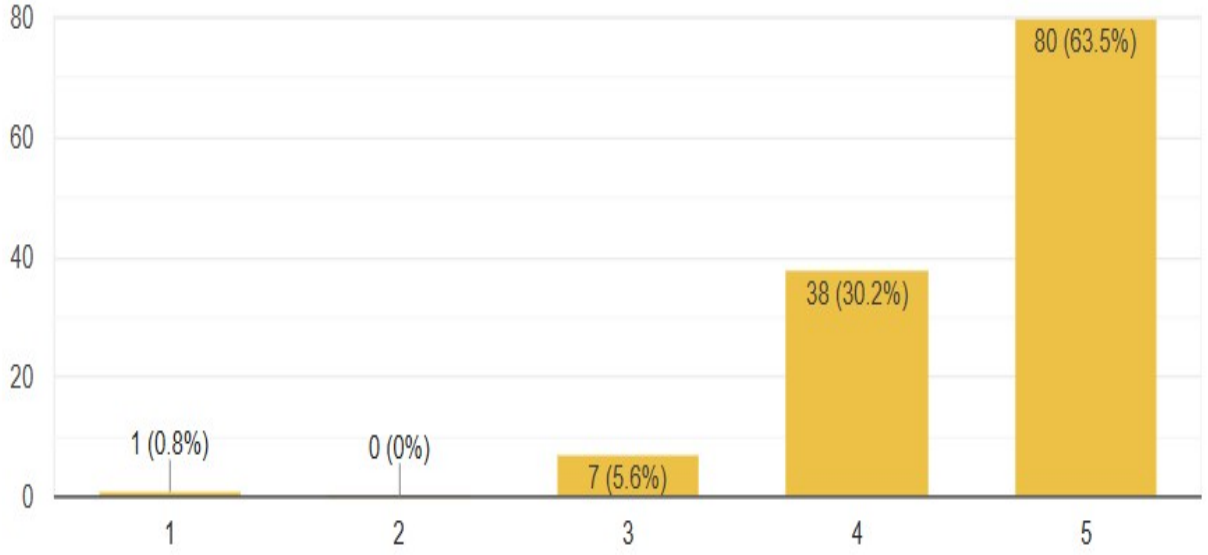
126 responses



How well did you understand Express Router, MVC architecture and CRUD operations?

Copy chart

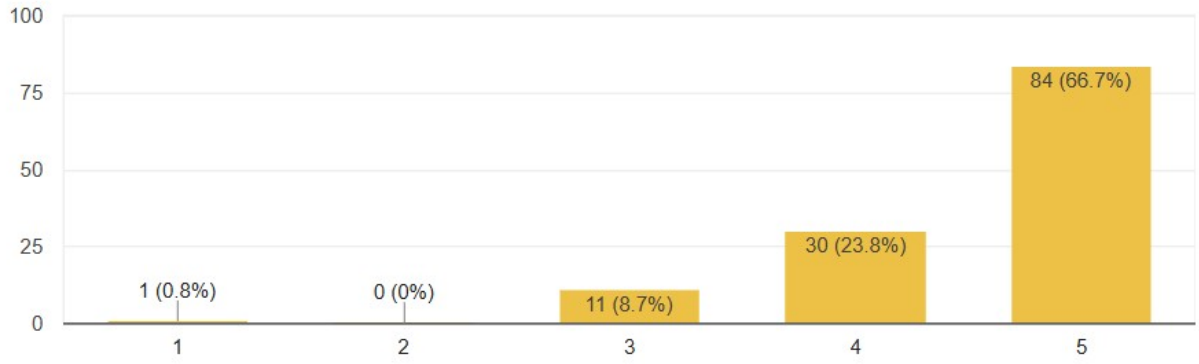
126 responses



How useful was the hands-on session on developing and testing Task CRUD APIs using Postman?

Copy chart

126 responses

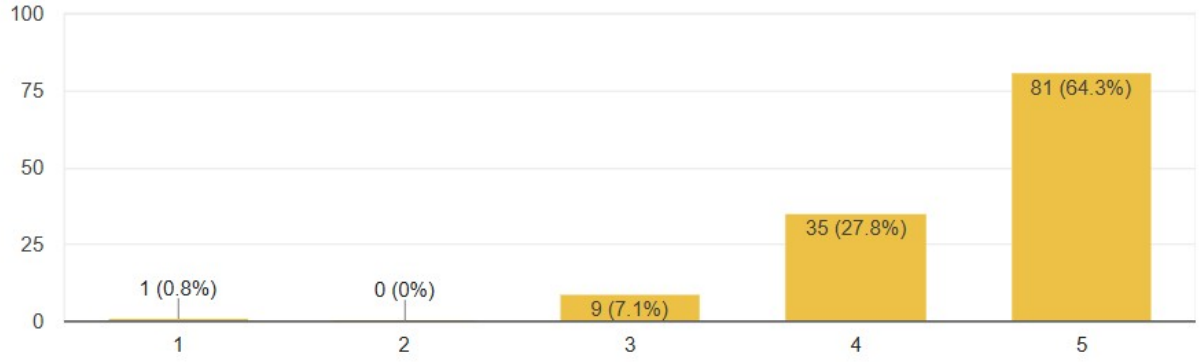


Day 6: Monday, 12-10-2026 – REST API Design, Validation and Error Handling

How effectively did the trainer explain REST API principles and HTTP status codes?

Copy chart

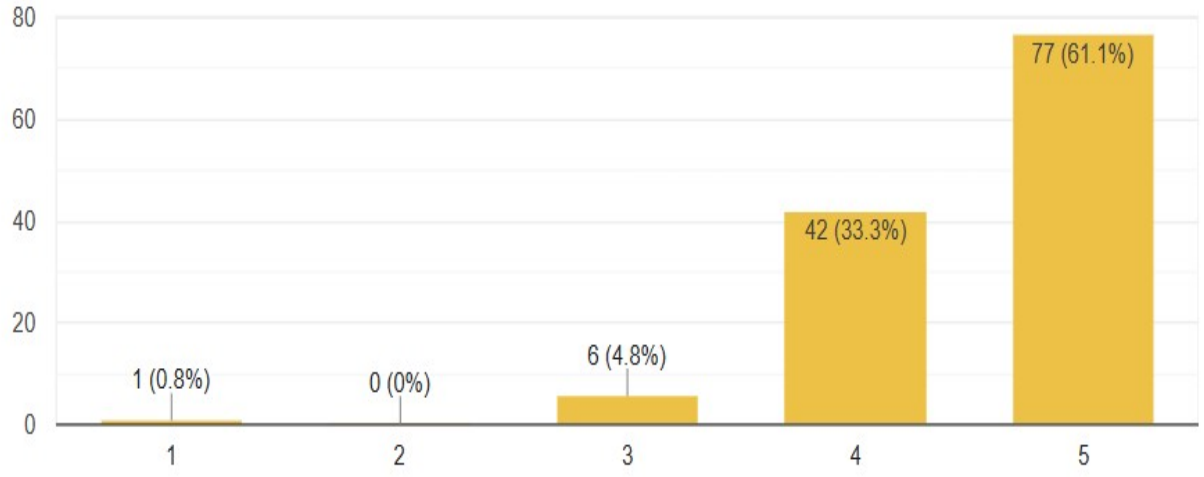
126 responses



How well did you understand input validation, centralized error handling and CORS?

Copy chart

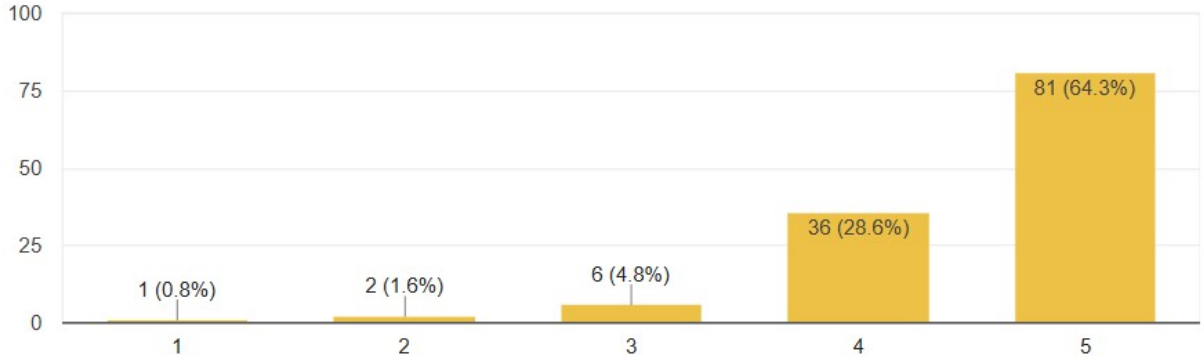
126 responses



How useful was the practical implementation of validated and error-handled REST APIs?

Copy chart

126 responses

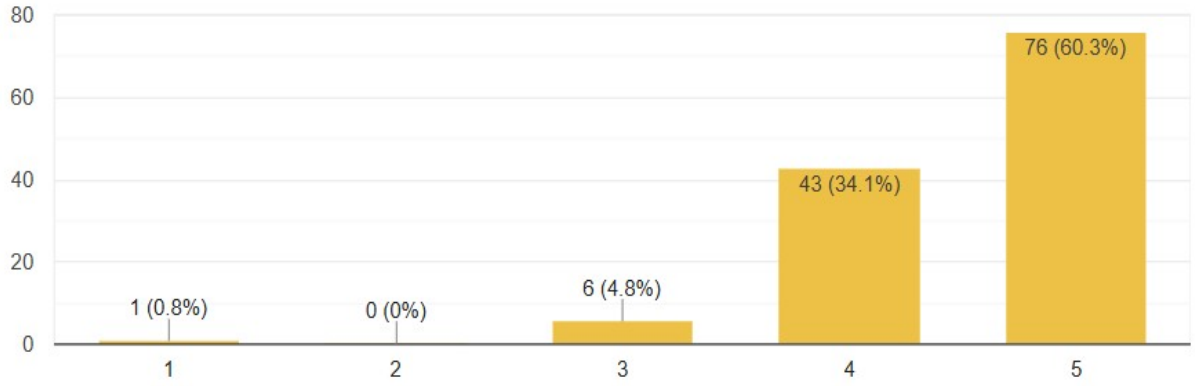


Day 7: Tuesday, 13-10-2026 – MongoDB Atlas and NoSQL Fundamentals

How clearly were NoSQL concepts, SQL comparison, documents and collections explained?

Copy chart

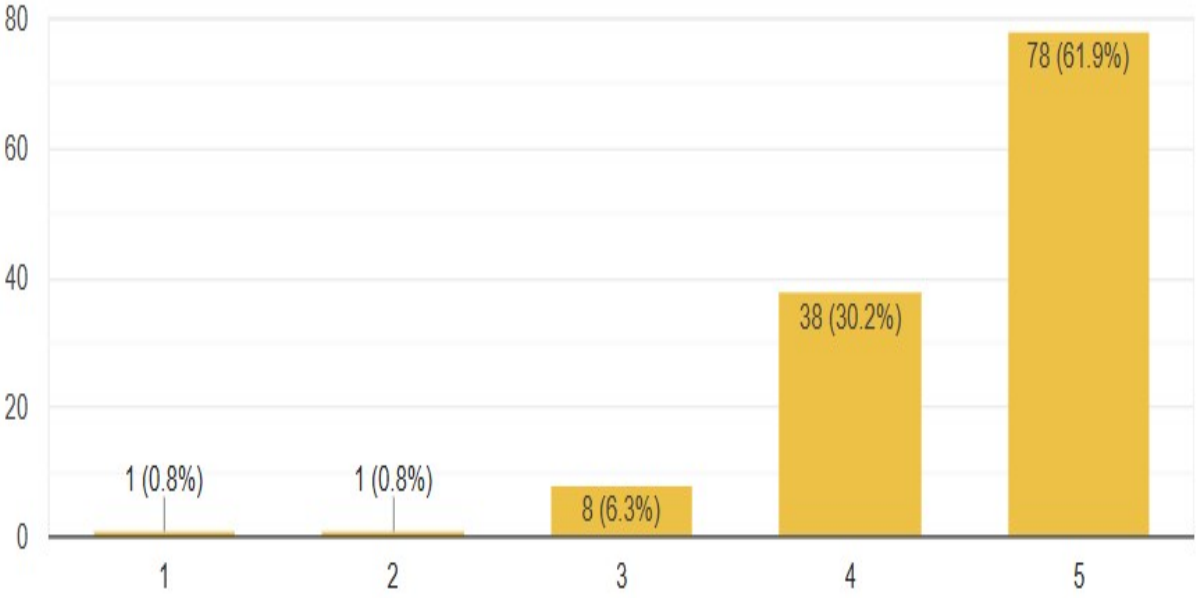
126 responses



How effectively were MongoDB Atlas and MongoDB Compass demonstrated?

Copy chart

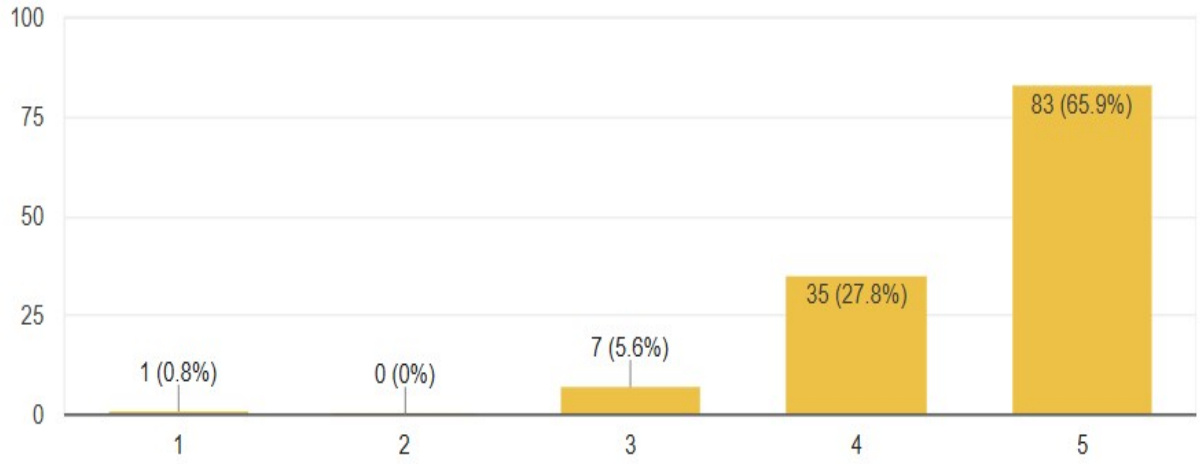
126 responses



How useful was the practical session on creating a cloud database and performing basic CRUD operations?

Copy chart

126 responses

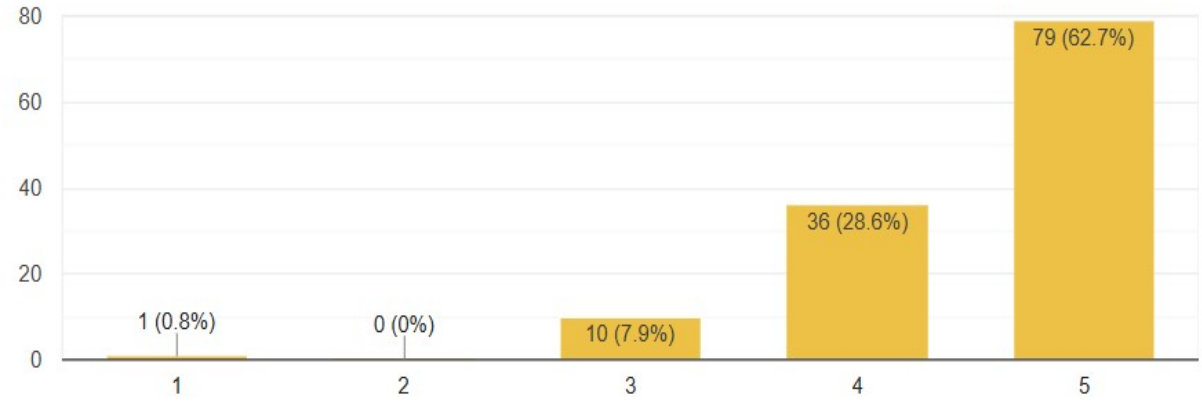


Day 8: Wednesday, 14-10-2026 – Mongoose Schemas, Models and CRUD

How effectively did the trainer explain connecting Node.js applications with MongoDB?

Copy chart

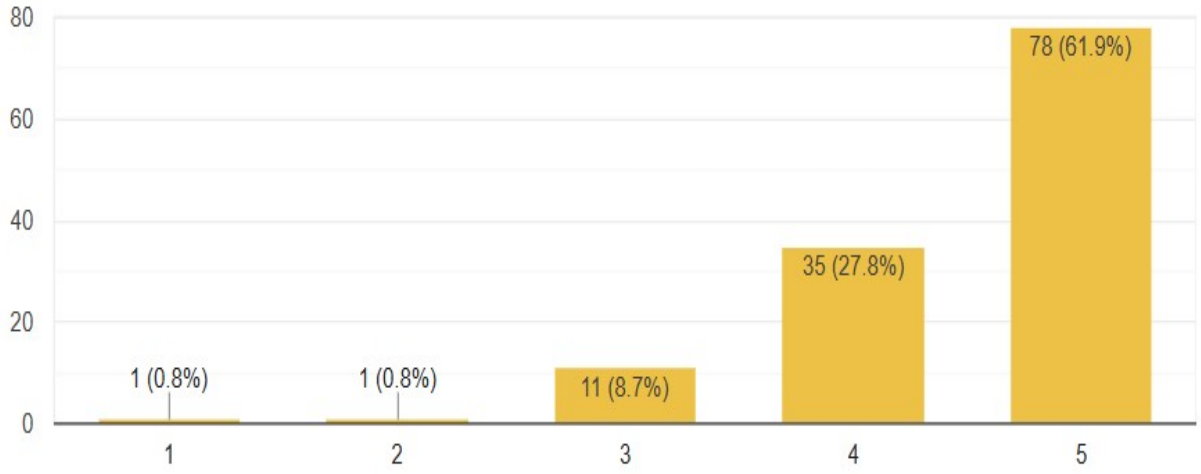
126 responses



How well did you understand Mongoose schemas, models, data types and validations?

Copy chart

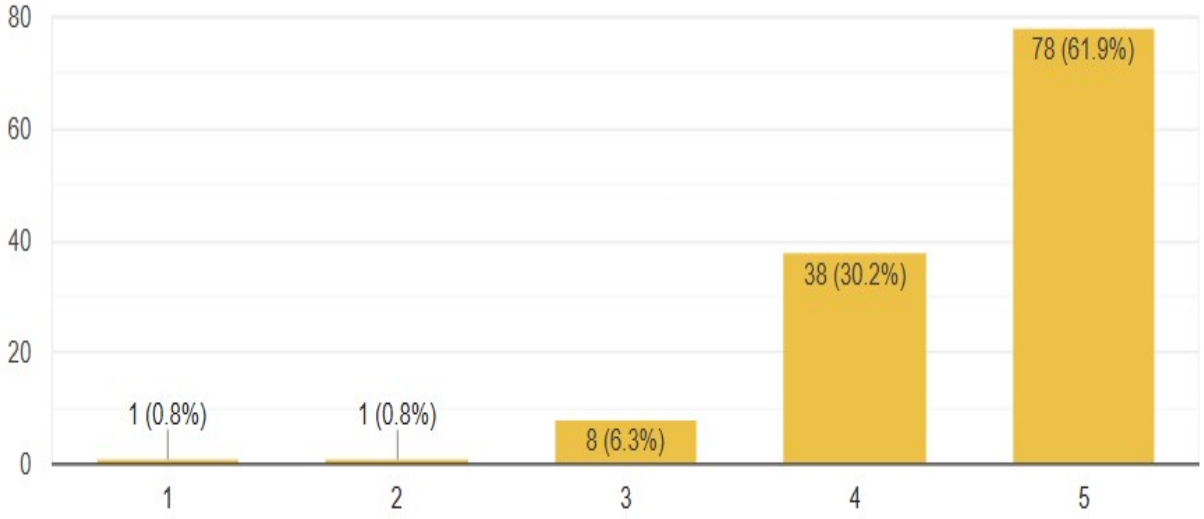
126 responses



How useful was the hands-on session on implementing CRUD operations using Mongoose?

Copy chart

126 responses

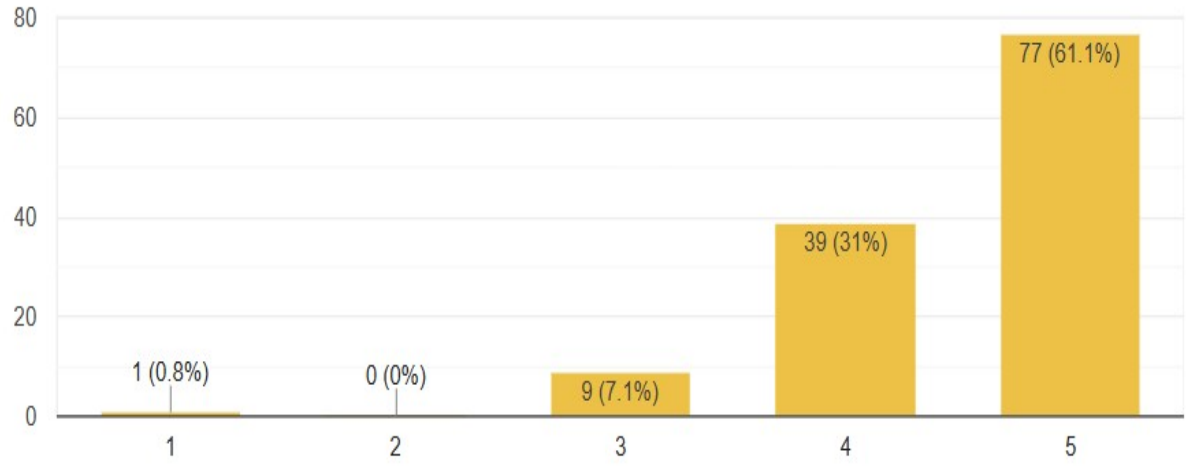


Day 9: Thursday, 15-10-2026 – MongoDB Queries and Relationships

How clearly were MongoDB query operators, filtering, sorting and pagination explained?

Copy chart

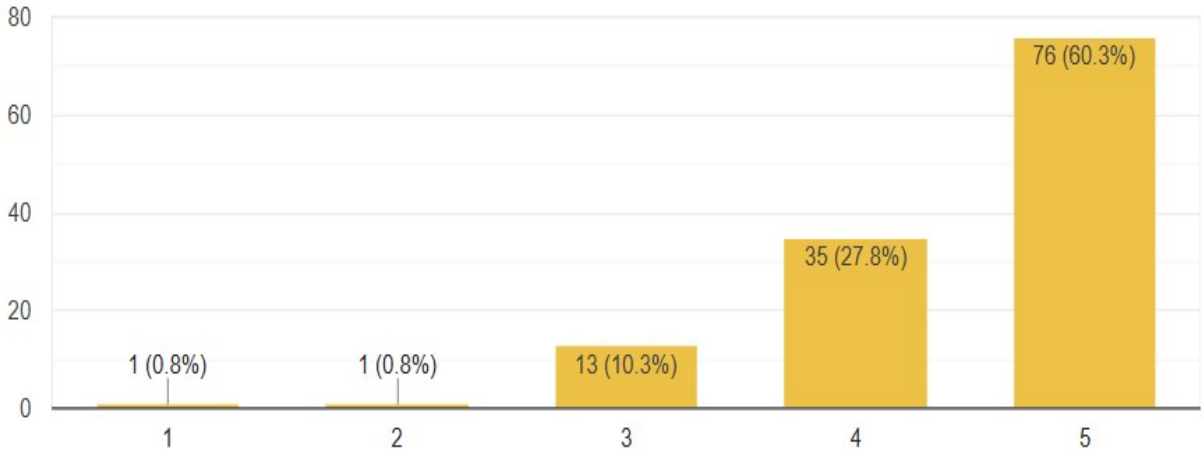
126 responses



How well did you understand document relationships using references and populate?

Copy chart

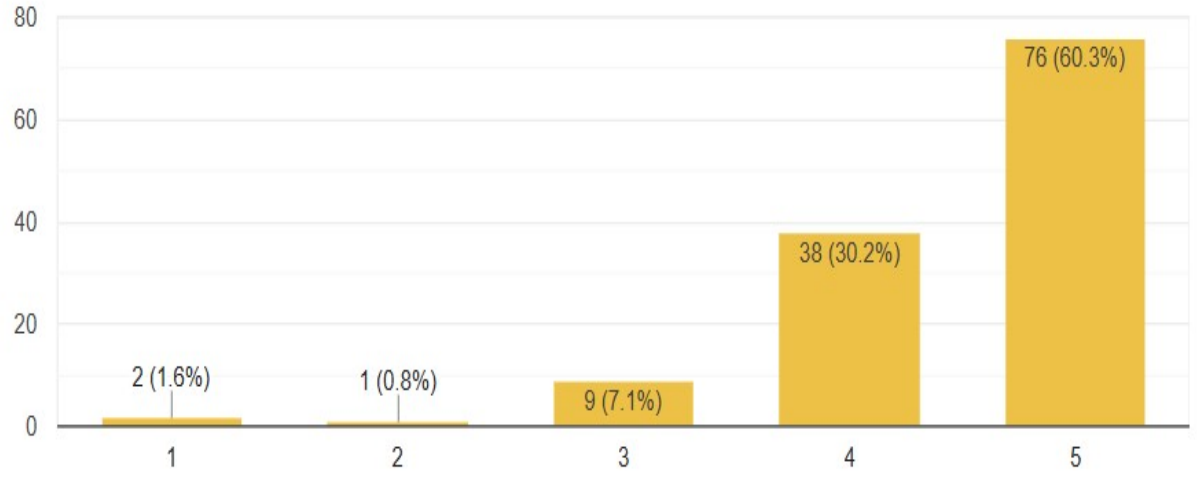
126 responses



How useful was the practical implementation of user-task relationships and filtering tasks by status and priority?

Copy chart

126 responses

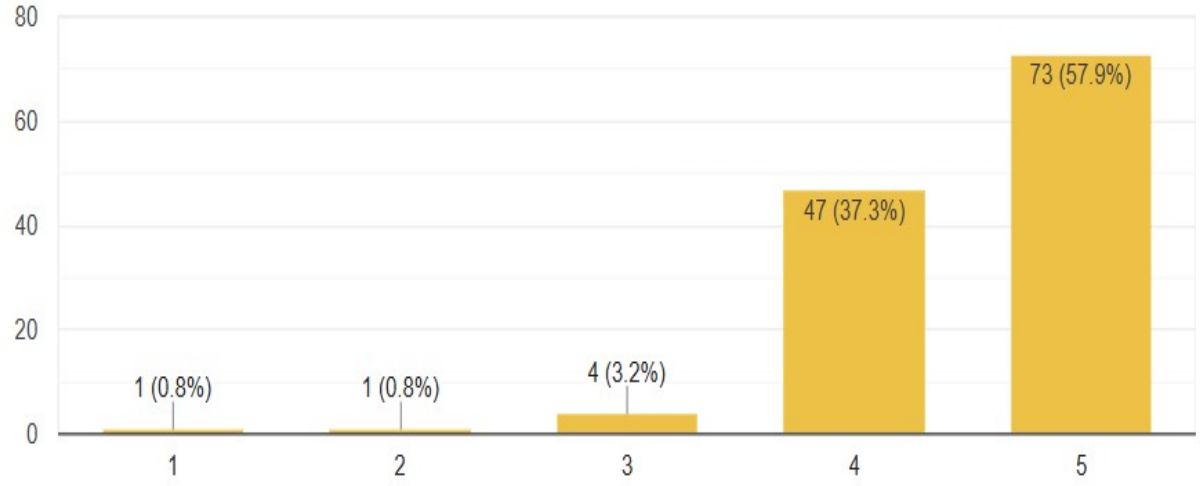


Day 10: Friday, 16-10-2026 – React.js Fundamentals and UI Components

How effectively did the trainer explain React.js fundamentals, SPA concepts and React versus JSP?

Copy chart

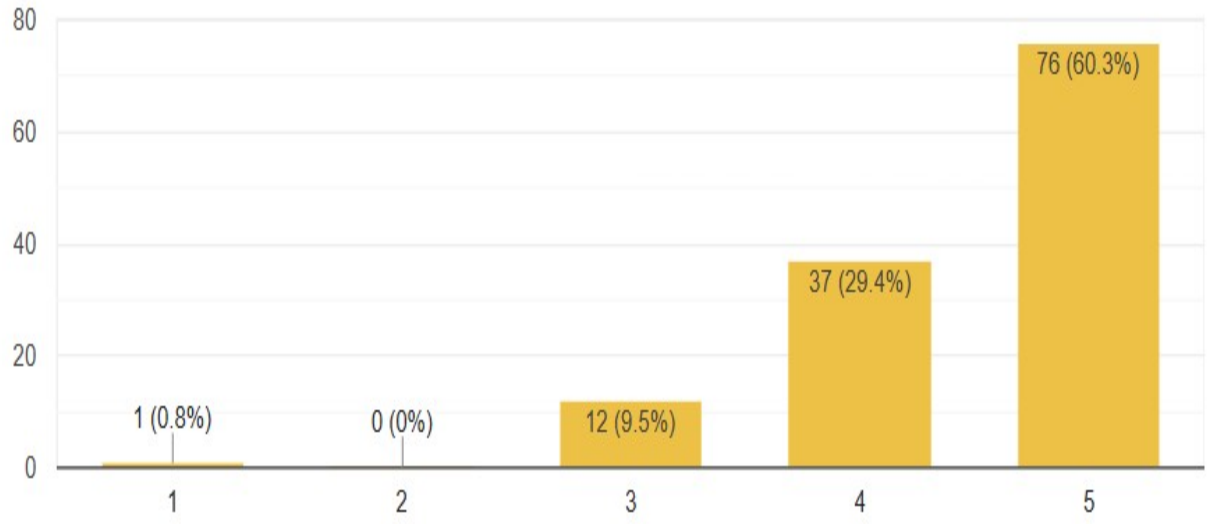
126 responses



How well did you understand Vite, JSX, components and props?

Copy chart

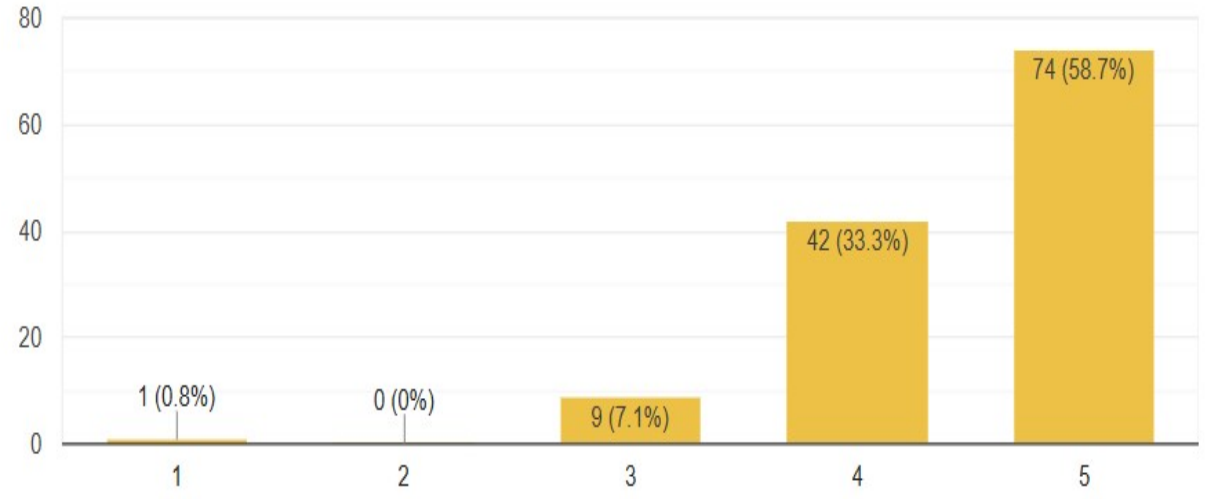
126 responses



How useful was the practical session on developing Task List and Task Card interfaces?

Copy chart

126 responses

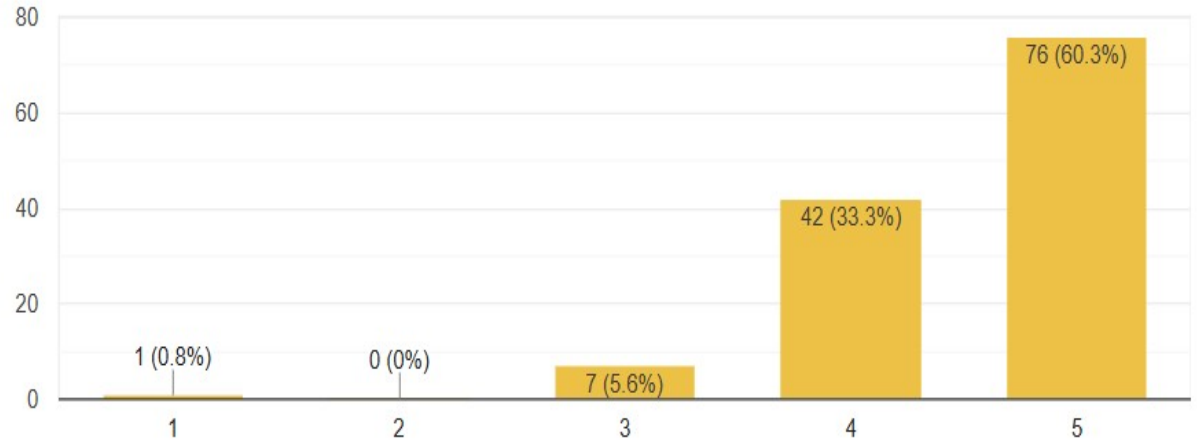


Day 11: Monday, 26-10-2026 – React State, Events and Forms

How clearly were useState, event handling and controlled forms explained?

Copy chart

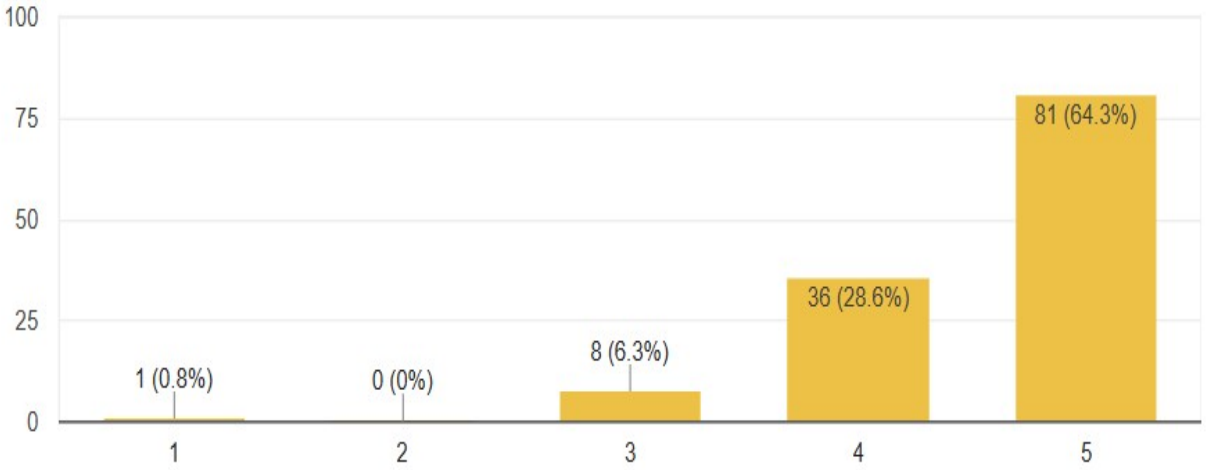
126 responses



How well did you understand list rendering, keys and conditional rendering?

Copy chart

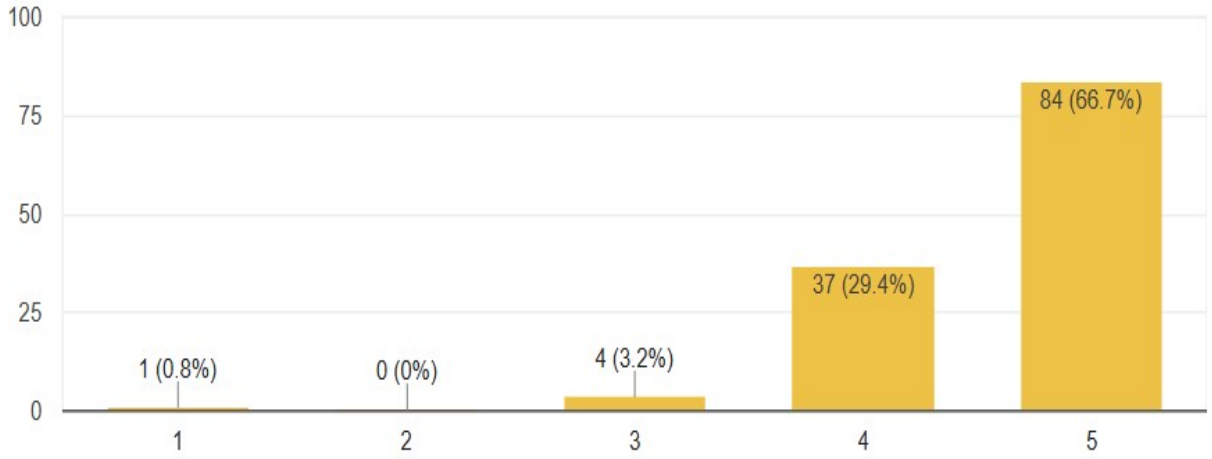
126 responses



How useful was the practical implementation of adding and deleting tasks using local state?

Copy chart

126 responses

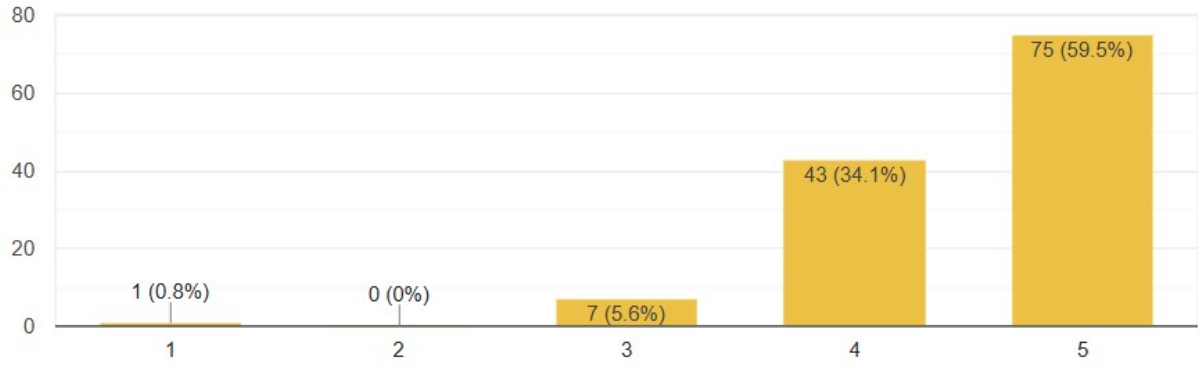


Day 12: Tuesday, 27-10-2026 – React API Integration

How effectively did the trainer explain useEffect and REST API integration using Fetch/Axios?

Copy chart

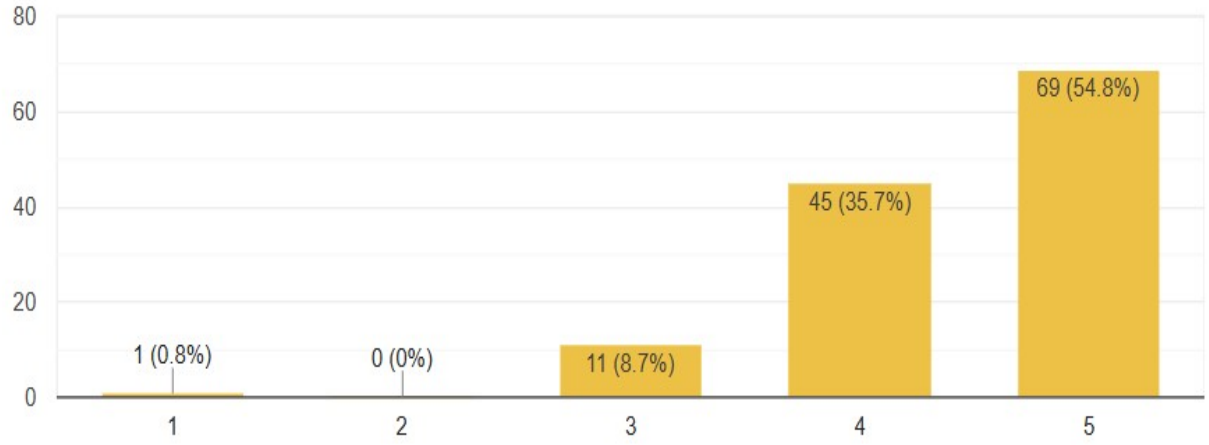
126 responses



How well did you understand loading states, error handling and frontend-backend communication?

Copy chart

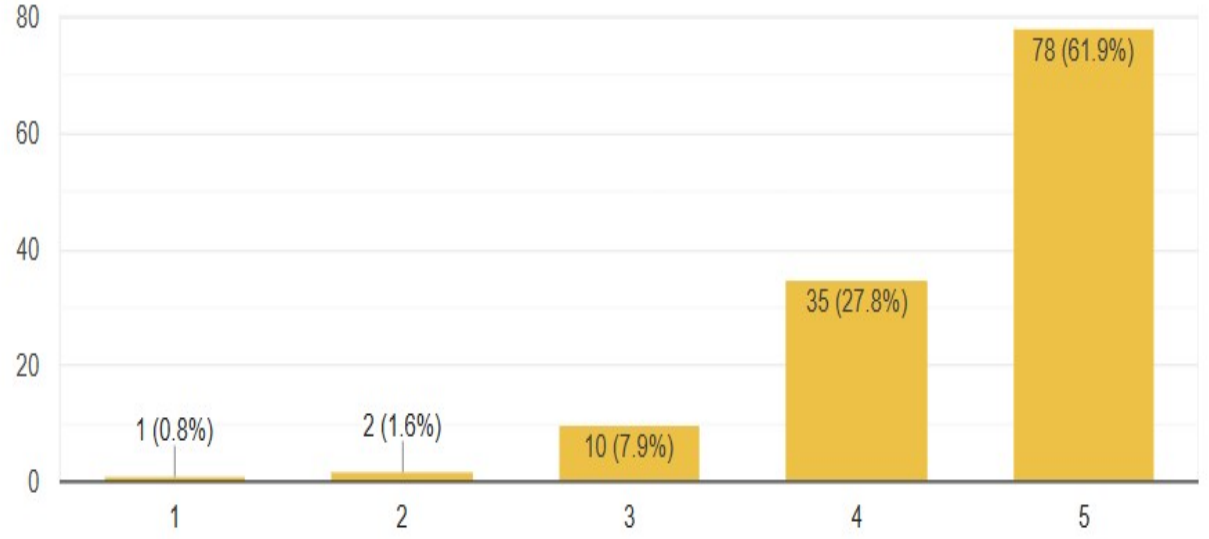
126 responses



How useful was the practical session on connecting React with Express APIs?

Copy chart

126 responses

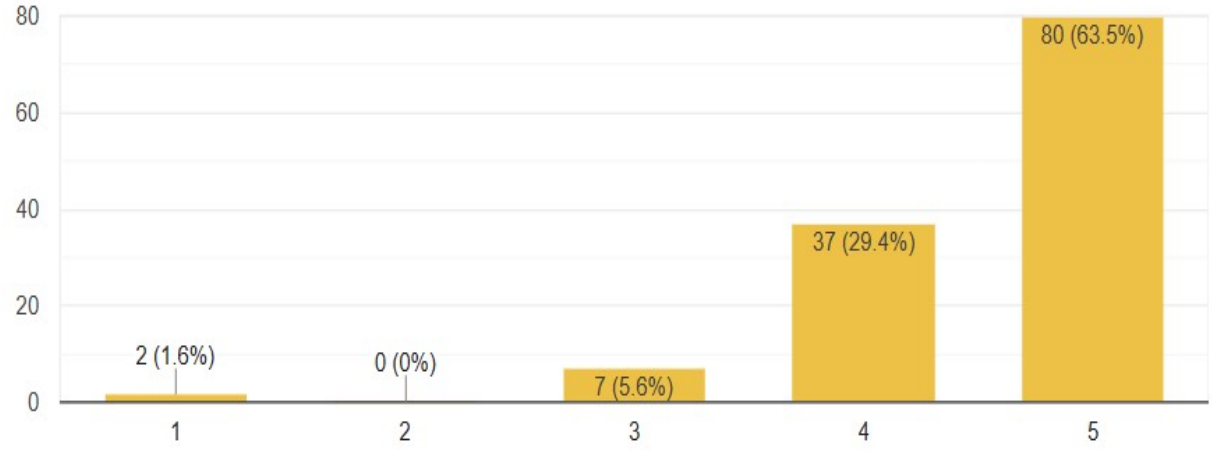


Day 13: Wednesday, 28-10-2026 – React Router and Full CRUD Integration

How clearly were React Router and multi-page navigation explained?

Copy chart

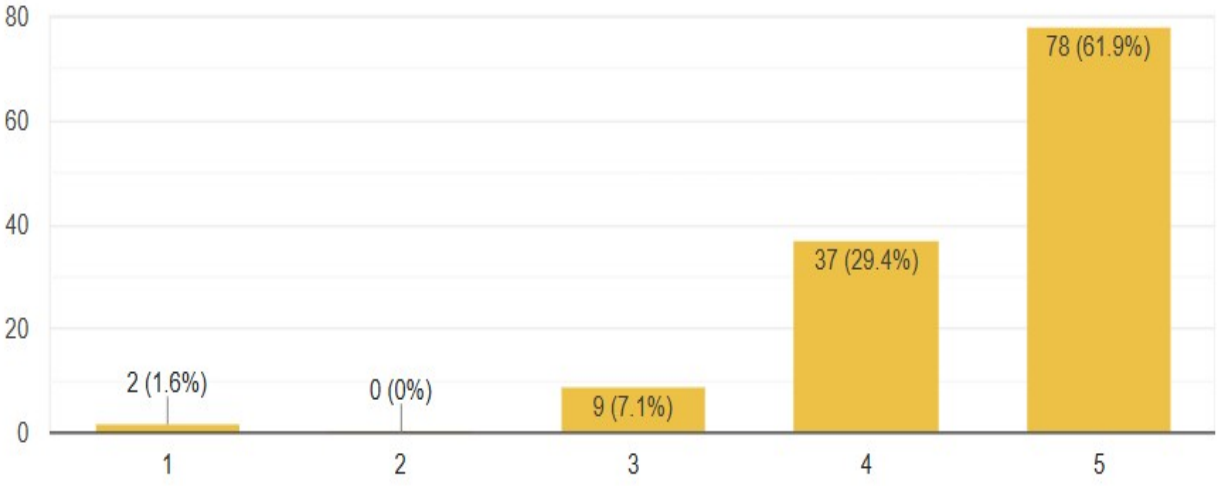
126 responses



How well did you understand state management and connecting frontend pages with backend APIs?

Copy chart

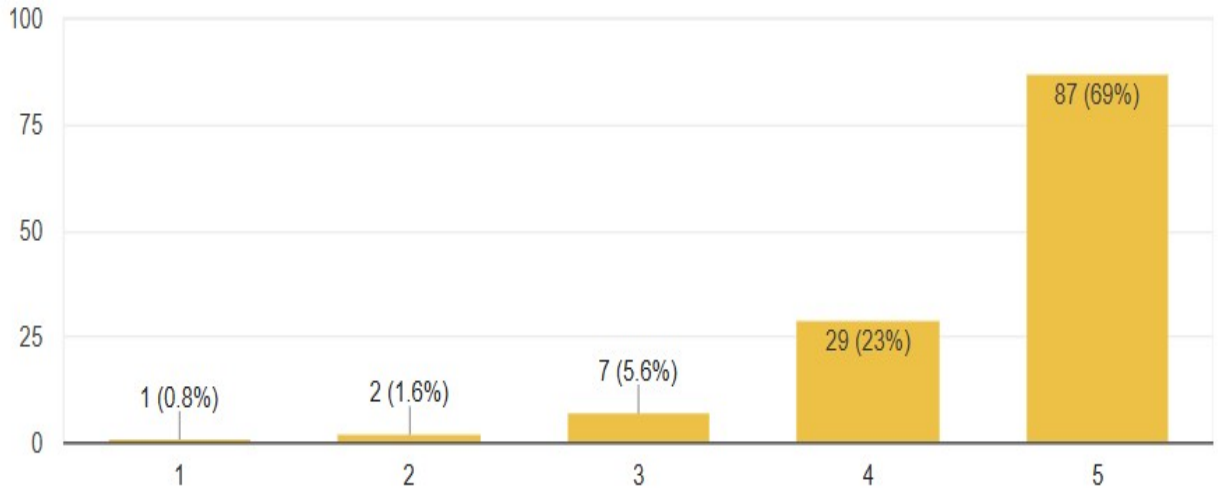
126 responses



How useful was the practical implementation of Task List, Add Task and Edit Task functionalities?

Copy chart

126 responses

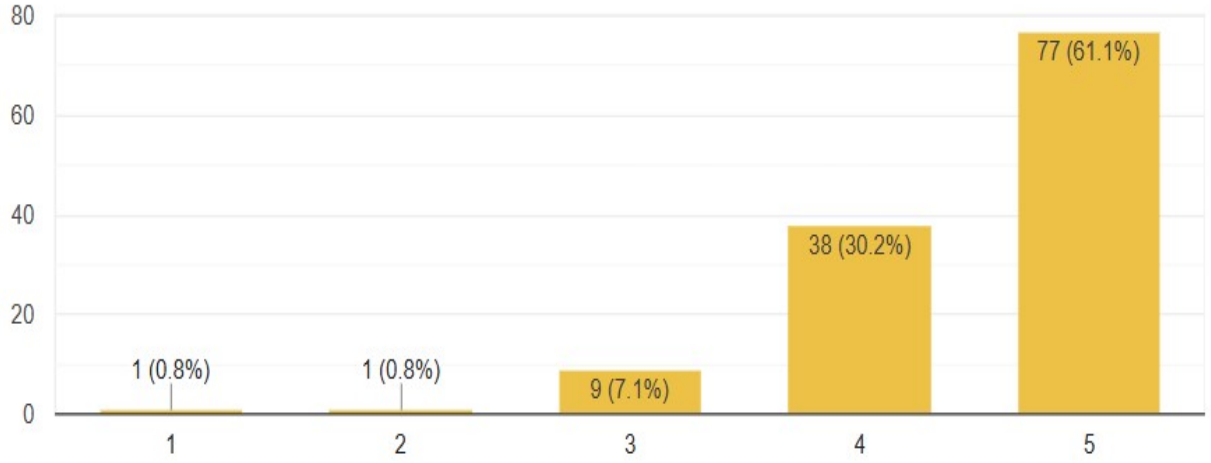


Day 14: Thursday, 29-10-2026 – Authentication and Protected Routes

How effectively did the trainer explain password hashing using bcrypt and JWT authentication?

Copy chart

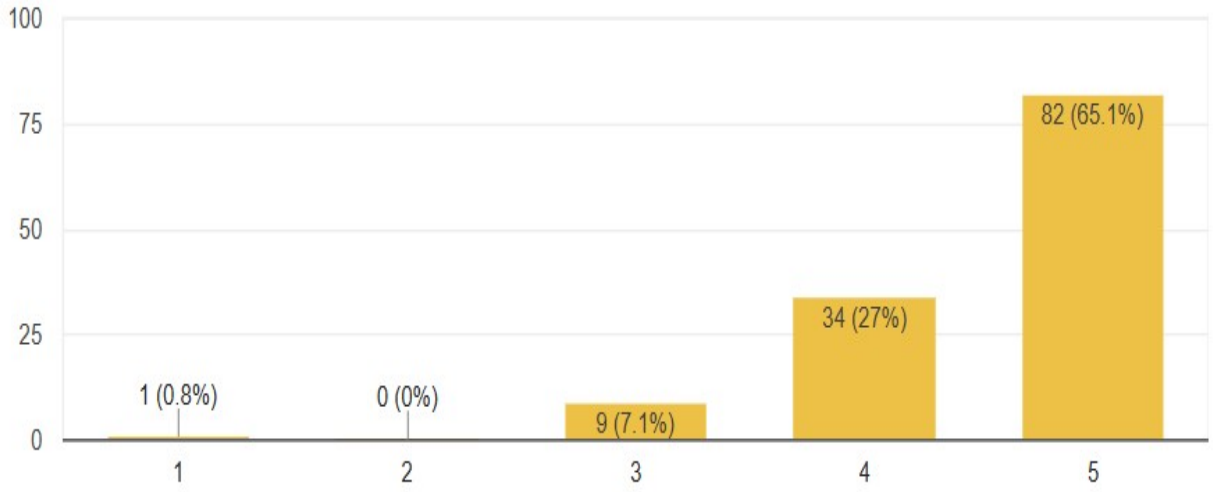
126 responses



How well did you understand user registration, login and protected routes?

Copy chart

126 responses

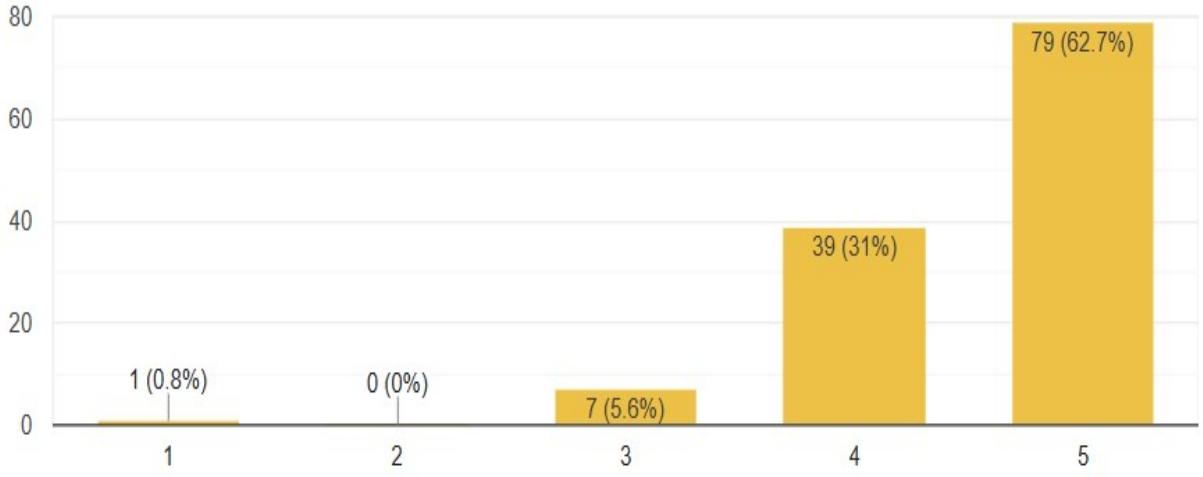


Day 15: Friday, 30-10-2026 – End-to-End Integration and Capstone Demonstration

How effectively did the trainer demonstrate end-to-end integration and testing of the MERN application?

Copy chart

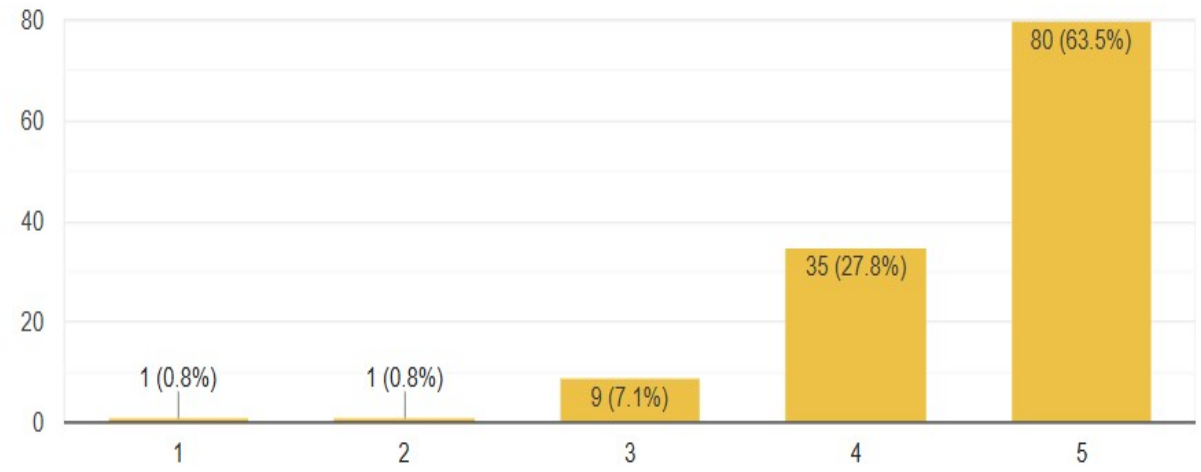
126 responses



How well did you understand the deployment process using Render and Vercel?

Copy chart

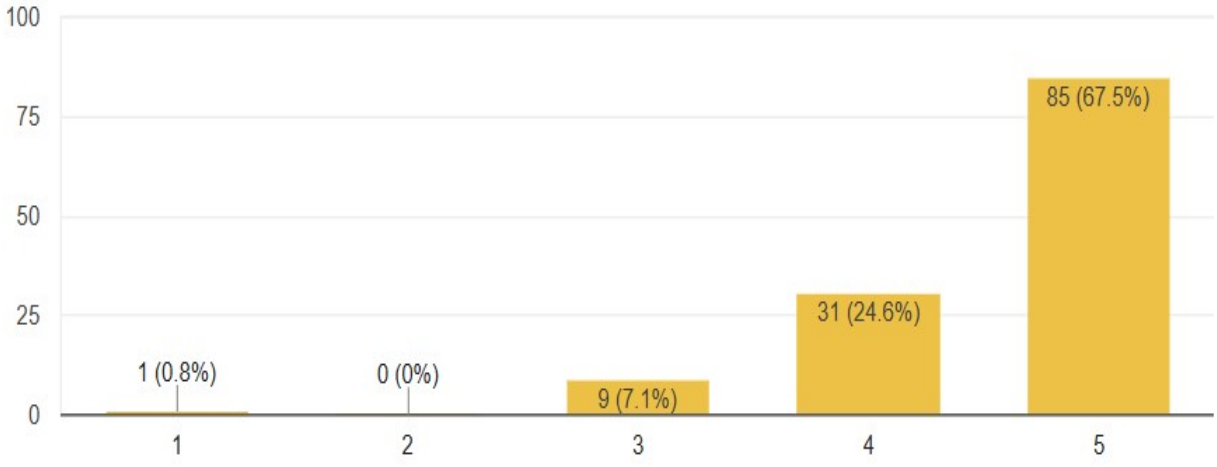
126 responses



How useful was the Student Task Manager capstone demonstration and final evaluation?

Copy chart

126 responses

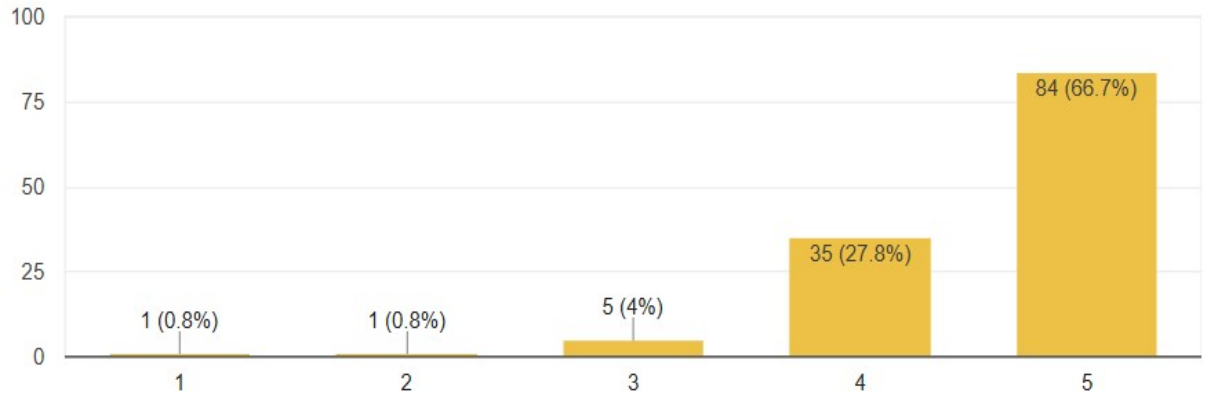


Feedback on the Trainer

How would you rate the trainer's subject knowledge and technical expertise in MERN Stack Development?

Copy chart

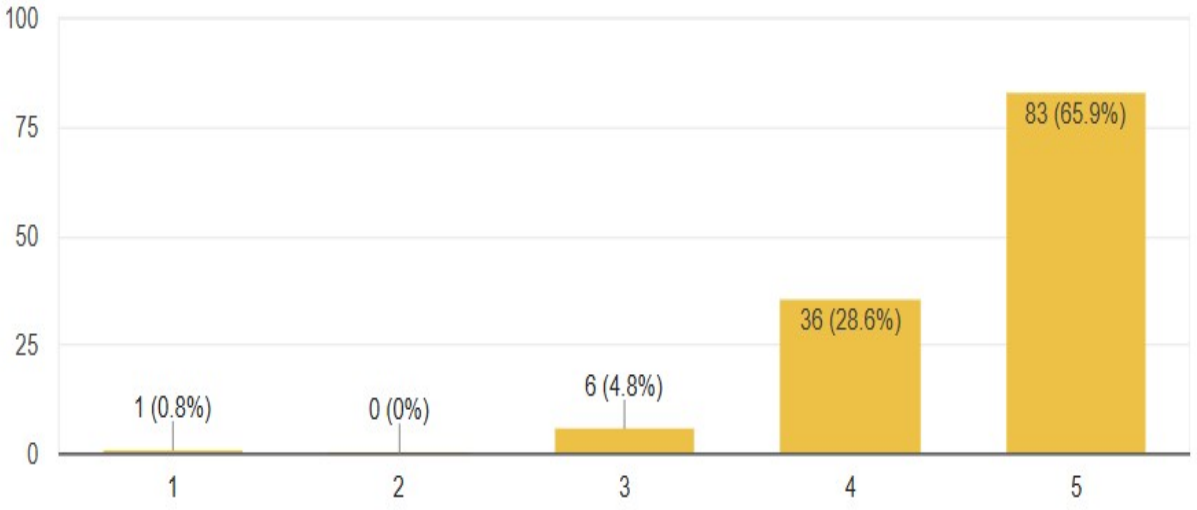
126 responses



How would you rate the trainer's teaching methodology, explanation skills and ability to clarify students' doubts?

Copy chart

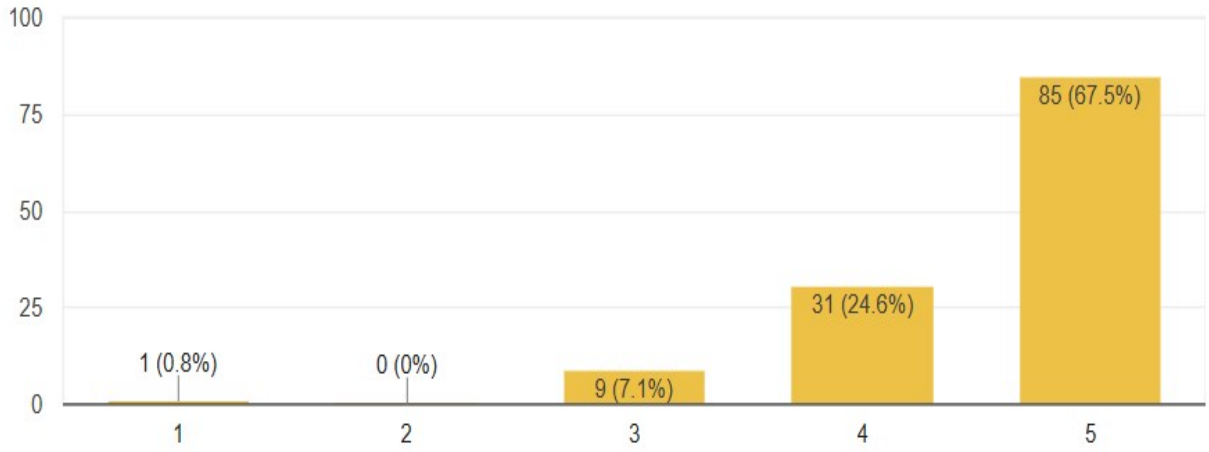
126 responses



How would you rate the trainer's interaction with students, time management and support during practical sessions?

Copy chart

126 responses

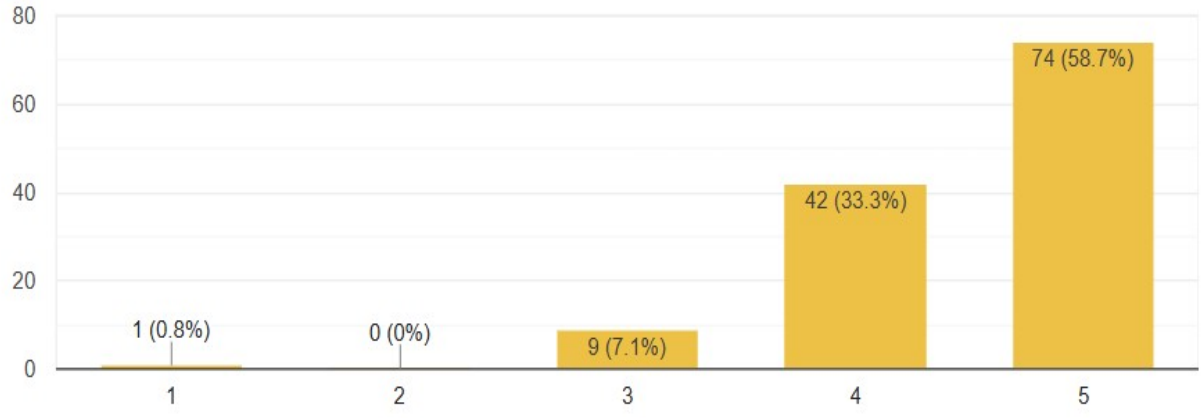


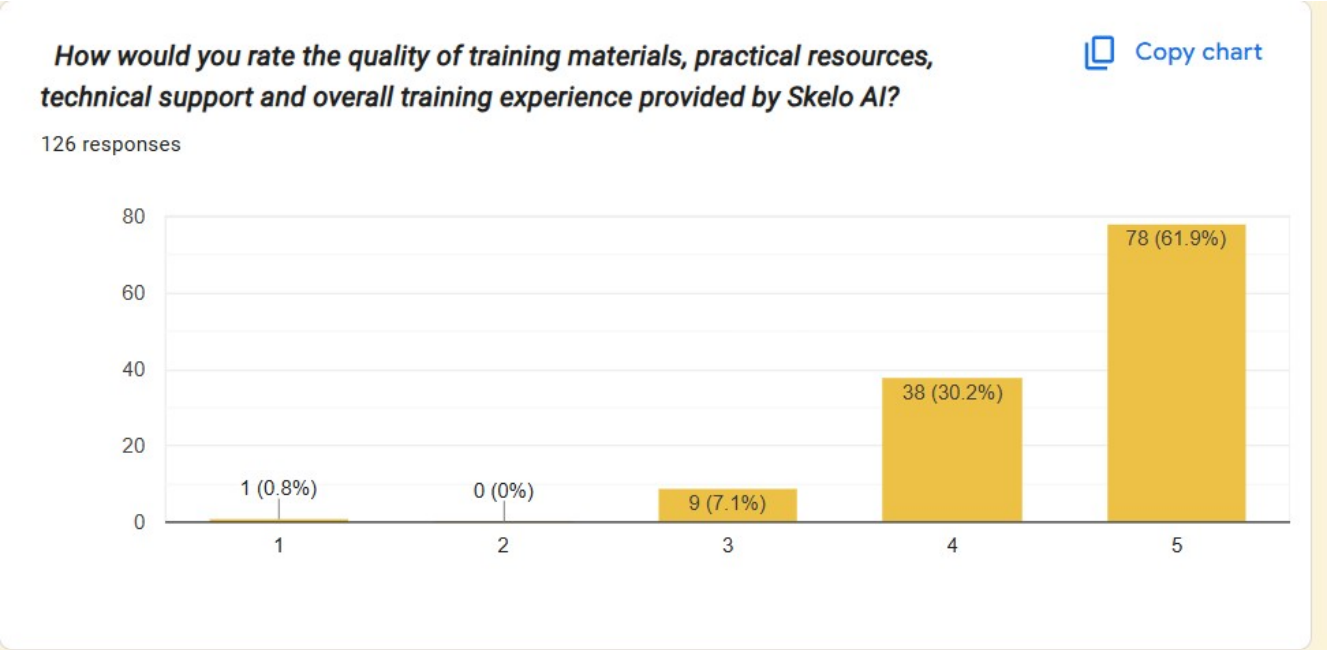
Feedback on Skelo AI Organization

How would you rate Skelo AI's overall organization, coordination and management of the 15-day MERN Stack Training Programme?

Copy chart

126 responses





10. OUTCOMES OF THE PROGRAMME

- Introduction to server-side application development using Node.js.
- Practical exposure to Express.js routing, middleware and REST API development.
- Fundamentals of MongoDB Atlas and Mongoose-based data operations.
- Exposure to React.js components, props, state, events, forms and routing.
- Guided integration of frontend, backend and database components.
- Introduction to authentication using bcrypt and JWT and to protected application routes.
- Curriculum designed to culminate in the Student Task Manager capstone demonstration and evaluation.

11. CONCLUSION

The 15-Day MERN Stack Hands-On Online Training Program, organised by the Department of Computer Science & Engineering, LBRCE, in association with Skelo AI, was conducted from 05 October 2026 to 30 October 2026 through Microsoft Teams. The programme followed a progressive learning path covering Node.js, Express.js, MongoDB, React.js and full-stack integration. The session-wise practical activities were designed around a Student Task Manager capstone application, providing students with an opportunity to connect FSD-II learning with modern full-stack web development practices.

CONVENER

Dr.S.NagarjunaReddy

Professor&HOD

LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING

(An Autonomous Institution since 2010)

Accredited by NAAC with 'A' Grade & NBA (Under Tier-I) | Approved by AICTE, New Delhi and Affiliated to JNTUK, Kakinada

L.B. REDDY NAGAR, MYLAVARAM, NTR DIST., A.P. – 521 230

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Mapping of Full Stack Development-II Course Outcomes with Sustainable Development Goals (SDGs)

FULL STACK DEVELOPMENT-II (23CSS4) | B.Tech. CSE – R23 Regulations

Programme: 15-Day MERN Stack Hands-on Online Training Program

SDG No.	Sustainable Development Goal	Justification / Relevance
SDG 4	Quality Education	Provides practical, hands-on training in MongoDB, Express.js, React.js and Node.js, enhancing students' technical knowledge and programming skills.
SDG 8	Decent Work and Economic Growth	Improves students' employability, industry readiness and career opportunities through full-stack development skills.
SDG 9	Industry, Innovation and Infrastructure	Encourages innovation and development of scalable web applications using modern technologies and software development frameworks.

Dr. S. Nagarjuna Reddy
Head of the Department

LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING

(An Autonomous Institution since 2010)
Accredited by NAAC with 'A' Grade & NBA (Under Tier-I) | Approved by AICTE, New Delhi and Affiliated to JNTUK, Kakinada
L.B. REDDY NAGAR, MYLAVARAM, NTR DIST., A.P. – 521 230

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Mapping of Full Stack Development-II Course Outcomes with Program Outcomes (POs) and PSOs

FULL STACK DEVELOPMENT-II (23CSS4) | B.Tech. CSE – R23 Regulations

The course outcomes of Full Stack Development-II are mapped with the Programme Outcomes and Programme Specific Outcomes as follows.

Full Stack Development-II Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
FSD-II-CO1: Apply JDBC and Servlet technologies to develop web applications that interact with relational databases.	2	2	2	1	2				1	1		1	2	2	3
FSD-II-CO2: Apply the MVC architecture using JSP, Servlets, and JavaBeans to design structured and maintainable web applications	2	2	3		2			1	2	2	1	1	3	2	1
FSD-II-CO3: Apply Node.js and Express.js to create RESTful services and perform backend operations with MongoDB.	2	3	3	2	3		1	1	1	1	1	2	3	3	3
FSD-II-CO4: Apply React.js to develop dynamic user interfaces and integrate them with backend APIs for full-stack web solutions.	2	2	3	2	3	1	1	1	2	2	1	2	3	3	2

Mapping scale: 1 – Low, 2 – Medium, 3 – High; blank – no significant direct mapping. PO1–PO12 and PSO1–PSO3 descriptions follow the LBRCE CSE R23 programme outcomes.

Dr. S. Nagarjuna Reddy
Head of the Department