COLLEGECODE:9623

COLLEGENAME:AMRITACOLLEGEOFENGINEERING AND TECHNOLOGY

DEPARTMENT:COMPUTERSCIENCEANDENGINEERING

STUDENT NM-ID: 12A034CC55E85E1BD8C491A91B2BC928

ROLL NO:962323104010

DATE:6-10-2025

CompletedtheprojectnamedasPhase5TECHNOLOGY

PROJECT NAME : Employee Directory with Search

SUBMITTEDBY,

NAME:R.N.AHILESHRAJA

MOBILE NO: 900356091

FinalDemoWalkthrough

1. LandingPage:

Displaysallexistingemployeesinatabularformat.

Search bar filters records dynamically.

2. LoginPage:

OnlyadminscanaccessCRUDfunctions. Implemented

using Flask-Login.

3. AddEmployeePage:

FormcollectsemployeeID,name,department,designation,email,phone. Validations

ensure data accuracy.

4. Edit/DeleteEmployee:

Admincanupdateexistingrecordsorremoveentries.

5. SearchBar:

Real-timefilteringusingAJAXrequests.

6. LogoutFeature:

Endssessionsecurelyandreturnstologinpage. System

Architecture

ArchitectureType:MVC(Model–View–Controller)

Layer Component Description

ModelDatabase (MySQL) Storesemployeerecords.

View HTML / CSS / JS Displays dynamic pages to user.

Controller Flaskroutes&API Handleslogic&communication.

DataFlow:

1. Userrequestsanactionvia UI.

2. Flaskroutesprocesstherequest.

3. MySQLq

4. ResponsesentbackasJSON→renderedonpage.

Backend Implementation

FlaskAppStructure:

/employee_directory

├ — app.py

├ —static/

├ —templates/

├ —models/

├ — routes/

├ — database.py

└—config.py

KeyAPIs:

GET /employees – List employees

POST/employees–Addemployee

PUT /employees/<id> – Update employee

DELETE/employees/<id>–Deleteemployee GET

/search?name=... – Search employee

LibrariesUsed:Flask,Flask-SQLAlchemy,Flask-Login,Jinja2

FrontendImplementation

HomePage:Displaysemployeetablewithsearchandpagination. Form

Pages: Bootstrap forms for adding/editing records.

JavaScriptFeatures:

Fetch API calls to Flask endpoints.

DOMmanipulationforliveupdates.

Validation before submission.

CSSDesign:Clean,minimal,responsivelayout.

Icons/UI:FontAwesome+Bootstrapiconsforclarity.

ScreenshotsDescription:

(Insertrealscreenshotslater–textplaceholdersbelow)

1. HomePage:Employeelisttablewithsearchbar.

2. AddEmployee:Formforaddingnewemployee.

3. EditEmployee:Formpre-filledwithcurrentdata.

4. DeleteConfirmation:Popuptoconfirmdeletion.

5. LoginPage:Adminloginform.

6. Dashboard:Overviewoftotalemployees&departments.

7. SearchResults:Filteredlistappearinginreal-time.

Testing andValidation

UnitTests:

VerifiedCRUDendpoints(POST,GET,PUT,DELETE).

Tested authentication and session flow.

IntegrationTests:

Checkedcommunicationbetweenfrontendandbackend APIs.

UITesting:

Manualtestingondesktop+mobilebrowsers.

PerformanceCheck:

Searchfunctiontestedon1000+records.

Result:

Allmodulesperformed withinacceptablelimits.

Challenges&Solutions

Challenge	Description	Solution
DatabaseConnectionErrors	Deploymentcausedenvironment issues	Usedenvironmentvariables +SQLAlchemypooling
SlowSearch	Largedatasetqueriesdelayedresults	Addedindexesandqueryoptimization
UI Scaling	Interfacebrokeonmobiledevices	UsedBootstrapgridsystem
Authentication Bugs	Session timeout & redirect issues	IntegratedFlask-Loginsession handler
APIIntegrationInconsistentJSONformats		StandardizedAPIresponseswithschemas

---

GitHubREADME&SetupGuide

SetupSteps:

1. Gitclone https://github.com/Ahilesh2005/Employee-directory-with-search

2. cdemployee-directory

3. python-mvenvvenv

4. venv\Scripts\activateorsourcevenv/bin/activate

5. pipinstall-r requirements.txt

6. ConfigureDBin.envor config.py

7. pythonsetup_db.py

8. pythonapp.py

Deployment:

Push to GitHub → Deploy on Render / Heroku.

Setenvironmentvariables(DB_URI,SECRET_KEY).

Test live link.

FinalSubmissionandConclusion

RepositoryLink:

https://github.com/Ahilesh2005/Employee-directory-with-search/tree/main

Deployed URL:

https://employee-directory-with-search.vercel.app/

Conclusion:

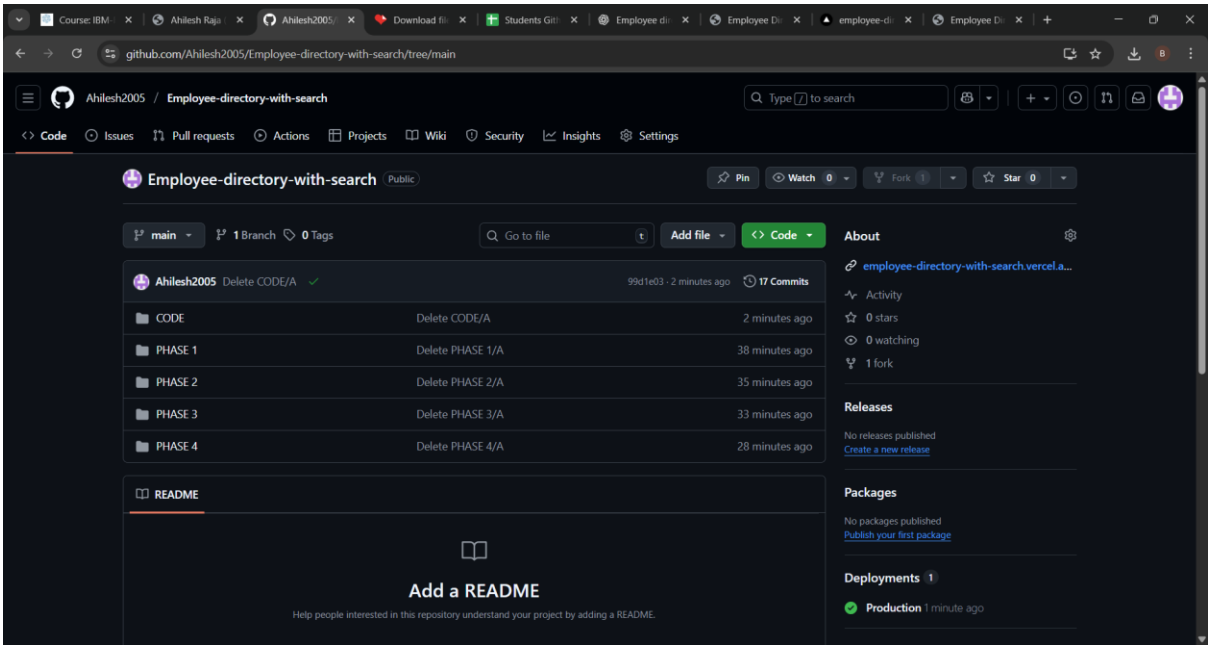
The project successfully meets the objective Of creating a searchable,scalable,and user-friendly employee directory.

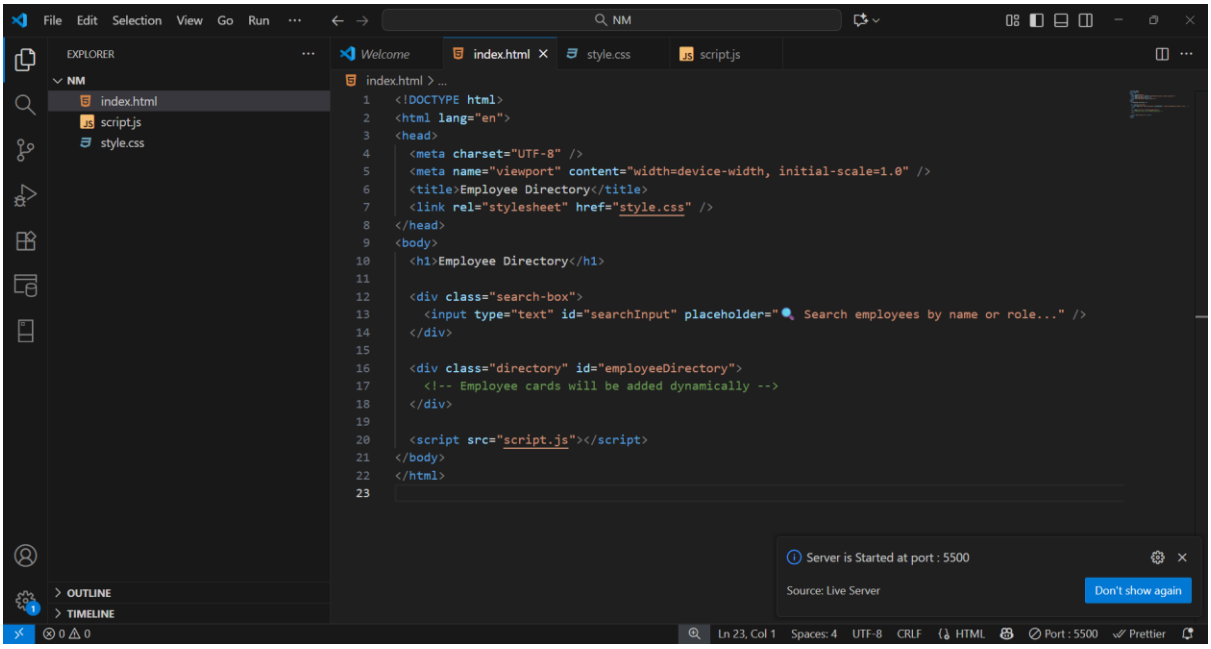
Through modular design and FlaskAPIintegration,it demonstrates efficient data handling and real-time interaction.

Future improvements include advanced filters,role-based access,and integration with external HR API.

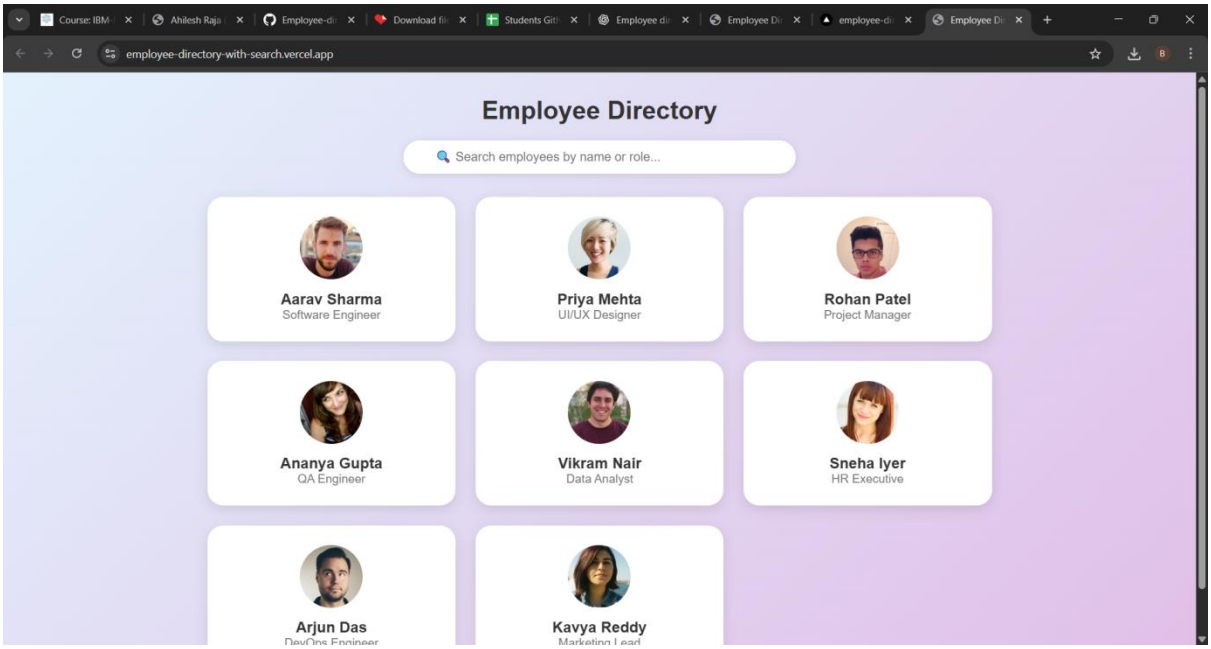
GITHUB: https://github.com/Ahilesh2005/Employee-directory-with-search

SCREENSHOT:





WEBSITE:


employee-directory-with-search.vercel.app
Employee Directory
Search employees by name or role...
Aarav Sharma
Software Engineer
Priya Mehta
UI/UX Designer
Rohan Patel
Project Manager
Ananya Gupta
QA Engineer
Vikram Nair
Data Analyst
Sneha Iyer
HR Executive
Arjun Das
Kavya Reddy