Name => Anjani Shaw.
Class => MCA 4th sem
Roll no. => 2305260004
Project Synopsis and Project Diary.

supremo

Project Title : Proconnect

Project Synopsis:-

Introduction:-

Proconnect is a Comprehensive freelancing and crowdsourcing platform designed to connect Clients with skilled freelancers across various domains. The platform aims to provide a seamless experience for users to post a Jobs, post a project, communication efficiently and process payment securely. With an integrated admin panal, ProConnect ensures smooth management of users, earning, service request, project request and overall platform activities.

Freelancing has gained immense popularity, enabling professionals to work remotely and businesses to access a global talent pool. Pro Connect seeks to bridge the gap between employers and freelancers by offering an innovative, transparent and efficient online workspace. The platform is designed to support both short term gig based tasks and long term contractual projects.

Objectives:-

The key objectives of Proconnects are -

- To creat an interactive and user friendly

supremo

freelancing marketplace.

- To enable clients to post Job requirements and hire freelancer based on Skills and experience.
- To allow freelancers to browse available Jobs and submit proposals.
- To develop a robust admin panel for efficient platform management.
- To implement secure payment processing and escrow services.
- To ensure transparency and trust through a review and Rating system.
- In Our website we have mentioned 3 dashboards they are Client dashboard, Freelancer dashboard and Admin dashboard.
- Specifies User Management and Freelancer management Panal in it.
- We have separate SignUP for User i.e Client and for a freelancer.

supremo

Features:-

User Module:-

Clients:-

- User friendly based to manage posted Jobs, and ongoing projects.
- Ability to review freelancer profiles, ratings and past work.
- secure payment option, In client dashboard we have added panals like MyProject, Service Request, Project Request, Profile, Review and Rating and lastly logout.

Freellancers:-

- Comprehensive profile creation with skills, experience and portfolio display.
- Secure fund payment upon project completion and approval.
- In our freelancer dashboard we have added pages like Projects, Earnings, Profile, Service Request, Project Complete Request and Logout.

Admin:-

- A centralized dashboard to monitor platform activity
- In Admin dashboard we have added

supremo

pages as category, User Management, Freelancer Management, Report and analytics and logout.

- In Admin panal we can add category according to his choice and this certaria is given by Admin.

Technologies Used:-

- Frontend : HTML, CSS, Java script.
- Backend : Java entity class, Hibernate, Thymleaf.
- Database : MySQL.
- Authentication : Spring boot.

Expected Outcome:-

Proconnect will streamline the freelancing process ~~will~~ by offering a secure, scalable and efficient platform where client and freelancer can connect and collaborate effortlessly. The project will enhancer the employment opportunities for freelancers while providing businesses with a reliable workforce.

The expected outcome include:

- An increase in Job postings and freelancer

supremo

engagement.
- A secure transaction machism ensuring out trust.
- Improved user experience with AI powered Recommendations.
- A well organized admin panal for smooth operation.

- Conclusion:—

Proconnect is a practical solution for Modern freelancing needs, offering a well structured workflow, secure transactions, and a user friendly interface. with its extensive features set and robust backend it aims to be a competitive platform in the freelancing industry.

By integrating advanced technologies User freelancer, admin, client dashboards.

This freelancing platform is not just an academic project but has real-world applicability, capable of serving the growing gig economy.

supremo

week-1 13/1/2025

Assigned The Project i.e A Freelancing Crowdsourcing website.
Name of our website - PROCONNECT.

System SetUP:-

1) Installation and system checking of VS code.
2) Installation & setup of Java, JDK.
3) Installation & setup of database (MysQL)
4) Environment setting of MYSQL.
5) Installation of Spring boot in VScode Extension.
6) Basic idea about spring boot.
7) Demonstration on spring boot project.

Advice:-

Adviced to visit 5-10 websites similar to our project content for referral the basic overview of our project on the topic we are going to design and implement the new ideas on it.

14/12/2025

- Creating zip folder on freelancing website project extraction of file.

- Creating home.html on template folder

- Creating the CSS file of it.

- Specifing functional Modules -
  - → User Authentication Module
  - → service category
  - → Freelancer Module
  - → Service Request Module
  - → Service Payment Module.

User Module:-

Administrator Module

Freelancer Module.

Client ~~Client~~ Module.

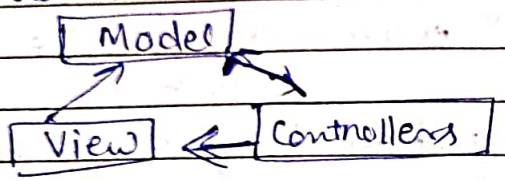
MVC:-

→ MVC stands for Model-View-Controller.

→ It is a software design pattern that separates an application's logic into 3 interconnected components: the Model (data and business logic), the view (user interface) and the controller (handles user

supremo

input and updates the Model and View.



Model:-

The Model component corresponds to all the data-related logic that the user works with. This can represent either the data that is being transferred between the View & Controller components or any other business logic related data. For example - a Customer object will retrieve the customer information from the database, manipulate it and update it data back to the data base or use it to render data.

View:-

The View component is used for all the UI logic of the application. For example the Customer View will include all the UI components such as text boxes, dropdowns etc that the final user interacts with.

Controller:-

Controller acts as an interface betn Model & View components to process all the business logic & incoming requests, manipulate data using the Model component & interact with the views to render the final output. Ex - The customer controller will handle all the interactions and inputs from the customer. View & update the database using the Customer Model. The same controller will be used to view the customer data.

- Desiging of our logo for our website

- Spring Boot:-

Spring Boot makes it easy to create stand alone, production-grade spring based application that you can just run.

Features:-

- Create stand alone spring applications
- Embed provide starter dependencies to simplify your build configuration.
- Automatically configure spring and 3rd party libraries wherever possible.

supremo
supremo

Designing of Project:- 15/1/25

- Desiging our Landing page i.e Home Page under Client folder in Template of Resources
- Write Main ProConnectApplication.Java file where the Projects runs.
- Creating the index.html page
- checking properties in application.properties file
- Mention the port = 1211
- spring.datasource.url = Jdbc:mysql://localhost/Proconnect.
- spring.datasource.username = root
- Spring.datasource.password = $(mysql.Password)
- spring.Jpa.Hibernate.ddl-auto = update.

Designing of Landing page:- 16/01/25

→ The Landing page serves as the first point of interaction for users.

→ Choicing the better color theme for our project.

→ Make sure it is Responsive design

→ Ensured the landing Page is Responsive and adapting.

→ Design header, Main content and Footer.

→ In Head or Negation bar - Sign UP sign-In, Contact us with search bar and Logo has been added in our website.

→ Add the hero section

→ Put a good background image

→ Added Team section on it where we mention all Leading team Members.

→ Added Testimonial section.

→ Added Contact section

→ Added designer section on it.

→ Added Make it happen with freelancer section.

Desiging landing page:- 17/01/25

→ Designed Footer section included contact information, social media links and quick Navigation option.

→ Created Page AboutUs which layed in our home page section.

→ Created New page called Our solution our solution.html where we empowering Freelancers & Clients and Why choose us section on it.

→ Created another static Page Howitwork Howitwork.html under client folder.

→ Designed the respective CSS & JS page of them simultaneously.

18/01/25

• Designing of Category and Freelancer Page

• Give certain criteria for perticular category.

• In Freelancer section we add all freelancer or we can say we list our all freelancers in their and creating the PLP and PDP for category as well as freelancer.

• PLP stands for Product listing page
• PDP stands for Product description page.

19/01/25

• Learn about Git hub.
• Github.com/username.
cmd - Git clone url + enter.

Run the System
Resource → application.properties
echo % mysql. Password %.
[If pass word not set]

Save.
• How to commit Git changes
• How to work on Git

supremo

Git config --global user.email
anjanshaw2003@gmail.com

git config --global user.name
Anjanishaw27.

In terminal Git Bash.

For collboration with team mate:-
Open Github login In your Account
↓
Go to settings
↓
Collaborator
↓
Push password
↓
Add people.
↓
Approve

Shrutis
20.01.25

Week-2

20/1/25

- DBMS - Database Management System
- Server - The physical or virtual machine where the DBMS and application backend code reside.

- Data storage:- The Physical storag
Ex - hard drives where the data is stored.

- Using Getter, Setter Method.
- Using data-th-each, data-th-text.
- Importing Java entity classes for backend functionality

- We have used Jakarta framework
It is a collection of services that help you write enterprise applications that run on the Java Platform.
- It provide the infrastructure often required for these applications so you that you can focus on core features and business Logic.

21/1/25

* why Java known as Jakarta.
→ The name refers to the largest city on the island of Java and also the capital of Indonesia, Jakarta.
* What is Jakarta in spring boot?
→ Spring Jakarta Servlet is a core component of spring.

Spring Boot embeds a servelet container in a cloud friendly way. The default container embedded in an application generated with the spring initializr is Apache Tomcat.

* What is Java entity:—
An entity is a lightweight persistence domain object. Typically an entity represents a table in a relational database and each entity instance corresponds to a row in that table. The primary programming artifact of an entity is the entity class, although entities can use helper classes.

22/1/25

* What is Entity framework in Java?
→ Entity framework (EF) is an open source object relational mapping (ORM) framework for ADO.NET.

→ Spring Boot @ Controller:—

→ Spring is one of the most popular frameworks that simplifies the development of robust, scalable and maintainable application.
→ Spring provides various feature such as dependency injection (DI), Aspect Oriented programming AOP and support for Plain Old Java objects (POJOs), making it a preferred choice for Java developers.

→ @ Controller Annotation in Spring Boot:—
The @ Controller annotation is a specialization of the @ Component annotation.
It is used to mark a class as a controller in a spring MVC application. Controllers are responsible for handling

incoming web requests, processing them and returning the appropriate response.

Key points about @ Controller:—

- It is typically used in combination with the @RequestMapping annotation to map web request to specific handler method.
- It can only be applied to classes.
- It is part of the Spring MVC frame work and is used to create web controllers.
- The @controllers annotation allows the Spring framework to detect the class as a controller during component scanning.

Steps to use @ Controller Annotation:—

- Create a simple spring boot project
- Add the spring web dependency.
- Create a controller package
- Create a Controller Class
  i.e @controller - This is used to specify the controller.
- @RequestMapping - This is used to map to the Spring MCV controller method.
- @ResponseBody - Ensures the returned value is sent as an HTTP response instead of resolving a view name.
- Run the application.
- Test the application.

23/2/25

Controller in Our Proconnect:—

1) Admindashboard Controller. Java
2) Category Controller. Java
3) Client Controller. Java
4) Clientdashboard Controller. Java
5) Freelancer Controller. Java
6) Freelancerdashboard Controller. Java
7) Logic Controller. Java
8) Payment Controller. Java
9) Project Controller. Java
10) Project Request Controller. Java
11) Service Request Controller. Java

supremo
supremo

24/1/25

→ To create a project using Springboot we need to start with start.spring.io or spring initializer with Maven Project.

→ Java language, 3.4.3 Spring boot version having project Meta data:
group - edu.rims
→ Artifact - Proconnect, Name - Pro Connect.
Packagename - edu.rims.pro.connect with Jar Packaging, 17 Java.
Adding dependencies like -

I) Spring web:-
Build web, including RESTful, applications using Spring MVC. User Apache Tomcat as the default embedded container.

II) Thymleaf:-
A Modern server side Java Template engine for both web and standalone environments. Allows HTML to be correctly displayed in browsers and as static prototypes.

III) Spring Boot Dev Tools:-
Provides fast application restart, Linkload and configurations for enhanced development experience.

IV) MYSQL Driver:-
MySQL JDBC driver.

V) Spring data JPA:-
Persist data in SQL stores with Java Persistance API using spring Data and Hibernate.

VI) Validation I/O: Bean Validation with Hibernate Validator.

VII) Lombok (Developer tools): Java annotation library which helps to reduce boiler plat code.

25/1/25

Learning About how to Inline the project in specific files and folders.

- In Resources folder we created many files and folders.
- In static folder we keep images in images folder.
- In Script folder we mentioned all necessary JS file of it.
- In Style folder we mentioned all the Css file in it.
- In Template folder mention 3 Section folder Admin, Freelancer and Client.
- All Admin Related page lies under Admin folder.
- All freelancer Related pages lies under freelancer folder.
- All Client Related Pages as well as Home page i.e Landing page and all other About us, How it work, Our solution page will be lie in client folder.

26/1/25

JDBC :-

→ JDBC or Java Database Connectivity is a Java API that allows Java application to connect to and interact with data base; enabling tasks like executing SQL queries and retrieving data.

→ It is an Application programming interface API for the Java programming language which define how a client may access a database. It is a Java based data access technology used for Java database connectivity.

- Create the Repository and Entity folder where you will mention all the Repository file & Entity Class with lombok.

Smruti
27.01.25

week-3

Git & Github:-

27/1/25

- Installation & download of Git & Github has already been done.

- How to Push code to Github.

- GitPull & Git Push commands to git & github

- Other feature including creating new branch and issue new ticket etc.

28/1/25

- Created files under the corresponding folder. Ex- html file, Css. file, image file, controller etc.

- Design the landing page of the project as a sample to get the view of the overall scenario.

- In body section of the Landing page contains the content of the platform's key features.

- Worked on designing the product listing page for the Pro connect website which include the details and name of the categories where the admin can edit & delete the categories.

- Worked on designing the PLP page with proper format for the listing page of the categories adding header section, navigation, search bar, footer section & logo.

- Worked on adding images in the category. Section with proper designing where the Client and freelancer can both browse categories for future transaction.

29/1/25

- Worked on designing the freelancer PLP page for the website where the admin & client can both visit to hire the skilled freelancer.

supremo
supremo

- worked on displaying the few details of the freelancer where the clients can easily make decision which freelancer is best for the project.

- Adding common features of the all the pages such as logo, header, navigation, footer using fragment.

- worked on adding images and features and skills of the freelancer where the user can easily distinguished between the freelancer skills.

- Worked on designing the product detail page or Pdp page for the freelancer profile where the freelancer can showcase their details such as name, address, experience, skill etc.

30/1/25

- Adjusting the Pdp page of freelancer style for better design adding images of the freelancer.

- Including the features such as header, navigation, footer.

- Learned about agile model, scrum Model & JDBC

- Introducing the branching of the webpages of the website.

31/1/25

- Learned about branching which is by default in master, we can create branch according to the webpage requirement.

- we can switch to the other branch from one branch.

- we can also commit the code using git push -u origin master or any other webpage for successful saving the code in the repository.

supremo
supremo

1/2/25

- We learned about Restrict the Master branch.

- Github: Settings → Rule → Ruleset → New ruleset → New branch.. Branch → Add classic → Branch name pattern: Master.

- Require a pull request before merging
  ↓
  Require approval → Create → Password.

VScode:-

Explore → Readme.md (file) → hello
(Ctrl+S)
↓
Commit → Git Push

Git hub:-
Setting → Branches → edit → do not allow by passing the above settings
↳ Save changes.

VS code:- Same activity in VS Code as before
↓
Commit → Git Push

supremo

2/2/25

Git Pull & Push:-

- How to merge in Master branch.
- Step by Step explanation.
- Generate Pull Request.-
  Pull Request → New Pull Request → Compair Master → home page → Create pull Request.
- Anjani → Request → Create pull Request
- Anjani: Pull Request → Add Your Review
  ↓
  Review changes
  ↓
  Approves
  ↓
  LGTM (Looks good to me)

- LGTM → Submit + Review → Merge Pull Request → Confirm merge.

- Changes → Stage block (+) → Git
  ↓
  Changes (+) → Commit.

supremo

mark down : marking with text style
-t ' ') backtick (left side of 1).

☑ viewed..

☐ comment : formate file..

(search)

- Shortcuts - VSCode → ctrl+P : filename
  Ex- line No: 273 → changes → git push

- Resolve conversation → Refresh → Request Review

Learn [Relative unit, absolute unit]
- cypress [Testing framework]

VSCode : Auto formate → settings →
Search : formate → ☑ formate & save.

- Default formatter file (Auto save).

Swathi
03.02.25

supremo

week-4

3/2/25

- Created table for each Java class or using Java entity class we created tables -
  - List dependent Table
  - Category least dependent
  Category → schema
  Database.

The column name should be same as Table Name.

- Added category Static Constant folder for each.

- Under Entity folder mention all Table class file:
  They are -
  1) Auditable.Java
  2) Category.Java
  3) Client.Java
  4) ContactMessage.Java
  5) Freelancer.Java
  6) Payment.Java
  7) Profile.Java
  8) Project.Java
  9) ProjectRequest.Java

supremo

10) Service Request . Java
11) User . Java .

. Auditing Class Contain 4 fields –
- ✓ Created-date
- ✓ Created-by
- ✓ Updated-date
- ✓ update-by.

. Make sure to extend auditable class in each Java Class to created date, created by, updated date, updated by.

. Category → image → Database
↓
folder
↓
S3 → AWS (on Demand Service)

supremo

4/2/25

Database file :–

- Create database folder under PROCONNECT.
- Create 2 .sql file they are –
  data.sql
  and script.sql.

- Script.sql :–

```
CREATE SCHEMA PRO-Connect
USE PROCONNECT
-- table
```

→ mention all the required file in it like user table, freelancer, category, project, payments, Reviews, Contact Messages.

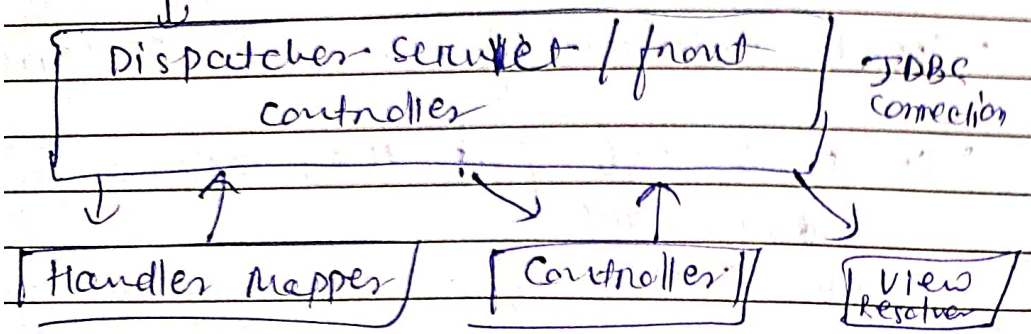
Data.sql :–

This file is used to store data that we will use in MyWork Bench to keep the same data or for referal we use data.sql file.

supremo

5/2/25

HTTP Request



Learned about UUID
universal unique identifier.

- Create enum class for category status
  Available, Unavailable.

1) CategoryStatus.Java
2) PaymentMethod.Java
3) Payment Status.Java
4) Project.RequestStatus.Java
5) Project status.Java
6) ServiceRequestStatus.Java
7) UserType.Java

- Learn about Relationship between the Modules.
  - one to one, one to many, many to one, Many to Many

- 1NF [comma separated]
  2NF [Partial dependency]
  3NF [No partial dependency]

one to many [alternative table we date for Primary Id.

Create user detail with user Role. enum class because enum classes are constant where we can store the user role which are client freelancer and admin.

- learned the Relationship E-E-R Entity-Entity Relationship how we can mapped the tables among each other.

- learned Cascading Restricted in MySQL if one table is Related

supremo

supremo

to other table then if we drop one table then the cannot be dropped unless the other tables are dropped as well.

- The mapped tables are showing the relationship which can be one to many or many to one or one to one or many to many.

6/2/25

- we learned how the MySql workbench work and how to execute Sql query in the schema.

- First we need to make schema where we can store all the tables related to the project data.

- we also learned to make, Java Entity class, enum class, auditing class etc which helps in data table.

The steps are -

1) Create a Java Entity class with Lombok.
2) Make separate enum class.
3) Make separate auditing class
4) Replace all Javax to Jakarta. import In auditable Java file we use @ setter & @ getter method.

- we can also set local data Time for future reference.

7/2/25

- Learned about Collection framework

Collection
- List
- Set
- Queue (Interface)

- Added fragment in our program created common.css file in it.

- Added dependencies
  web + dev + thymleaf

- <header data-th- replace="~{home::header}">
  <footer data-th- replace="~{home::footer}">

- Introducing Collection framework i.e. List, set, queue.

- In list duplicates are allowed, insertion order preserve, Random access

- List are of following types i.e arraylist, vector, linked list.

- In set, duplicate data are not allowed, insertion order preserve, searching can be O(1).

- Queue can be FIFO Type.

- Mapping can be pair of data that is stored as key value pair.

supremo

3/2/25

- Create fragment for common code in all webpage where the logo, header section, navigation bar, footer section are common.

- we use fragment because of reuse ability.

- In fragment we can replace all the code repeated in every webpage.

- we can write as in home page or any other page related to the fragment.

9/2/25

- Created tables for the project user table, Category table, project table, freelancer table, payment table etc.

- Introducing getter & setter method, JDBC connection, JPA, ORM.

10-02-25

supremo

week-5 10/2/25

- Introducing CI-CD workflow i.e continuous Integration, continuous deployment.

- Java object $\xrightarrow{\text{mapping}}$ Table row data.

- ORM i.e Object Relational mapping use hibernate (java), ibetics, sequelizer (node JS)

- Difference between JPA & Hibernate.

JPA:-

1) It is described in Javax.persistancer package.

2) It describes the handling of relational data in Java application.

3) It is only a Java specification.

4) It is a standard API that permits to perform database operations.

Hibernate:-

1) It is described in org.hibernate package

2) Hibernate is an Object Relational Mapping (ORM) tool that is used to save Java objects in the RDBMS.

3) Hibernate is an implementation of JPA. Hence the common standard which is given by JPA is followed by Hibernate.

4) It is used in mapping Java data types and database tables.

11/2/25

- Autowired in spring framework
@ Autowired is annotation used for Automatic dependency injection

- Allowing Spring to inject dependencies into your Classes by type, eliminating the need for manual configuration

- Field Injection: Annotate a field with @ Autowired.
- Constructor Injection: Annotate a

supremo
supremo

Constructor with @ Autowired.

- Setter Injection: Annotate a setter method with @ Autowired

- MVC pattern [@ Autowired
- Constructor based
- setter based

- Using List to bind data using find all ();

- Using Model attribute to bind the data

12/2/25

- Created the Repository class. Java files They are -

1) CategoryRepository. Java
2) ClientRepository. Java
3) FreelancerRepository. Java
4) PaymentRepository. Java
5) ProjectRequestRepository. Java
6) ServiceRequestRepository. Java
7) UserRepository. Java

JPA Repository:-

JPA Repository is a JPA-specific extension of the Repository interface, providing a convenient way to manage data persistence and retrieval by automatically implementing basic CRUD operations, pagination and sorting, reducing boilerplate code.

- JPA Repository provide method like save (), findbyId (), findAll (), deleteById () etc for common database operation.

Use JPA Repository:-

- Define a Repository interface. Create an interface that extends JPA Repository and specify the entity type and its ID type.

Enable JPA Repositories:-

Use the @ EnableJpaRepositories annotation in your configuration class to enable JPA repository support.

Inject the Repository :—
Inject the Repository interface into your service or controller class using spring's dependency injection.

Use Repository methods :—
Call the methods provide by the JPARepository interface to interact with the database.

13/2/25

- Use of sementic tag.
- ~~Sementic~~ Ticket generation.
- Issue fixing
New Issue → Add a title → Bind backend → Category ~~data~~

description → get data from database bind it on frontend home screen to Map the category.

- Ticket assign yourself.

Ticket table (bug Tacked)
documentation
duplicate
Enhancement.

- Development Create new branch Create on main (master) branch

git fetch origin
git checkout.

Commit msg #10
Category data bind on home page
git push.

Similar App as Github were Bid bucket, Git Lab

14/2/25

- Branch name should be short as CI/CD issue will come.

- Relationship security -
data export, data import.

- Data export -> Proconnect -> Start export
- Beekeeper App to store the data.

15/2/25

- Worked on MySQL workbench where we can export data between collaborator.

Steps are -
Server -> option -> select database
Export Command -> Start export

- For successful exporting data using a application i.e Beekeeper studio.

16/2/25

- Working on Payment Page, Registration Sign In Page.
- Also worked on Category PIP page.
- working on category table.

Swarnali
17.02.25

week-6

17/2/25

- Write code for logout
- Icon added in home page
- learn about UUID universally unique Identifier which are 128 bit labels used to uniquely identify objects in computer systems.
- Implement it on entity class as
- @ Generated Value (Strategy = UUID

- Worked on database table of category freelancer page along with its backend using post Mappings & get Mappings.

18/2/25

- Worked on sign up page designing with its html, CSS and js file.
- Also worked on developer or freelancer page.

19/2/25

- Worked on Pdp page of the category page.
- Fixing the issue in the category page connection.

20/2/25

- Worked on dashboard page and enhanced some design, searching table features for client and freelancer.

21/2/25

- Learned post mapping Concept Form : get & Post method where get show User name & password & post will not show.
- Used form method Post in Client Sign up page.

supremo

- Created freelancer dashboard.

22/2/25

- Worked on admin and client & freelancer dashboard.
- Adding category page in admin dashboard.

23/2/25

- Worked on dashboard page and trying to bind data from backend for our freelancer section.

Suruti
24.02.25

supremo

24/2/25

Week-7.

- Learned thymleaf concept, search concept using getmapping
- URL encoding which doesn't allow space in url.
- we need to create list of category in CategoryRepository
  i.e
  List<Category> findBy CategoryName Containing IgnoreCase (String CategoryName)

25/2/25

- Learned to create and upload image where the images are added from the frontend and store in the database using form enctype = multipart/form data"

<input accept = "image/png" image/jpg"> also connected using postmapping.

@RequestParam @ModelAttribute
CommandLineRunner.

26/2/25

- Worked on backend part of the dashboard page.
- Addition of upload images in the gitignore.
- Added CommandLineRunner ProConnectApplication.Java file

```
@Override
public void run(String args) throws
exceptions
{
    File file = new File("Pathname:"
        upload-images);

    if (file.exists())
    {
        file.mkdir();
    }
}
```

27/2/25

- Worked on category page of admin dashboard.
- Introducing UUID concept and where it implemented.

28/2/25

- Addition of form in category page of Admin dashboard.

1/3/25

- Worked on image uploading in admin dashboard.
- Implementation of UUID concept in the image uploading.

4/3/25

- Learned WORA concept, thread wrapper class.
- Worked on JPA Repository in the category

Suresh
3.03.25

Week-5

3/3/25

- Worked on freelancer dashboard page
- Worked on data binding on the category page.
- Addition of images in project and category page.

supremo
supremo

4/3/25

worked on adding data on the project page of the client dashboard.

- Addition of search query
  @ controller
  @ Request Mapping
  @ GetMapping ("/search")
  String search (@ Request param String search)

5/3/25

- Worked on backend and part of the freelancer page.
- Learned spring security in login, logout, session management.

Authentication :—
It is a method of verifying the identity of a consumer or system to ensure they are who they claim to be.
It involves checking credential which include username, password.

Authorization :—
It is the method of figuring out and granting permission to a demonstrated user or system, specifying what assets they can access and what actions they are allowed to carry out.

6/3/25

- Worked on linking frontend of the dashboard with the backend.
- Introducing encryption, decryption and bcryption formate where the plain text convert to hash.
- It also improves password strength.

Security Config :—
@ Bean.
PasswordEncoder encoder ()
{
return BcryptPasswordEncoder ()
}

supremo

- Addition of encoder in the login.

- Addition of UserDetails Service by using User Repository.

7/3/25

- Worked on search bar and logout option added.

- learning addition of Thymleaf security 6 dependency.

- @AllArgsConstructor. @NoArgsConstructor.

- Difference among warehousing, marketplace supplier, dropshipping

- Addition of UserService. Java.

8/3/25

- Worked on addition of PDP page and addition of pages in other dashboard.

- Addition of earning page, payment page changes in its content.

9/3/25

- Worked on Linking myproject and project description page.

- worked on Profile page in client dashboard.

10.03.25

supremo

week-9

10/3/25

- Working on adding data in table and payment page.

- MyProject Page and create another sign-in page one for client i.e user and another for freelancer.

11/3/25

- Worked on sign-up page.
- worked on PLP and PDP page of the freelancer page.
- Addition of data from backend to the frontend.

12/03/25

- Addition of data in admin & Client dashboard.
- Added service Request page.

13/3/25

- Working on UserManagement page and Freelancer Management.
- Styling and using fragment and handling CSS.

14/3/25

- Worked on Service Request Page and Project Request page in the Client dashboard.

15/3/25

- worked on service Request Page and Project completion Request page in freelancer dashboard.
- Backend databinding has been already done during this section.

supremo

16/3/25

→ Addition of Profile page in Freelancer dashboard.

→ Addition of Profile page in Client dashboard.

→ Binding data from backend has been done as well.

17.03.25

~~Week~~

week-10

17/3/25

- Addition of Status and action of the freelancer dashboard.

- Addition of the table in the Project Request.

18/3/25

- Create dtoRepository under the repository folder to extend JPA Repository.

→ Writing code for the edit button for the category and Project page.

19/3/25

→ Working of edit button functional and worked on the delete button of the category and project page.

20/3/25

→ Worked on search button or bar where we trying to search for category, projects and freelancer profile.

21/3/25

→ Working on spring security for sign in page for Authentication.

22/3/25

22.03.25

supremo

supremo