

PROJECT TITLE - PROCONNECT

Introduction :-

ProConnect is web-based freelancing platform designed to seamlessly connect clients in need of services with freelancers offering their skills and expertise. It aims to be a go-to digital space where professional collaboration happens efficiently, securely and without unnecessary complexity.

The concept behind ProConnect is to simplify and streamline the freelance hiring process. Clients often struggle to find trustworthy and skilled professionals, while freelancers face difficulties accessing quality projects that match their skill set. ProConnect bridges this gap by providing a structured Category-driven, and role-based platform that ensures both parties can connect, communicate and collaborate effectively.

ProConnect emphasizes a simple yet powerful user experience, backed by features such as project categorization, user authentication, and a responsive interface.

The platform enables clients to post jobs, freelancers to apply, and admins to manage the entire system ensuring that the platform remains organized, secure and scalable.

The development of ProConnect is not only a technical demonstration of full-stack web development skills but also a real-world solution to modern employment challenges. It encompasses the application of web technologies, database management, user role systems, CRUD operations, and intuitive front-end design to deliver a complete freelancing portal.

The system supports three primary user roles:

1. Admin - The platform administrator who manages project categories, monitors user activity, ensures content moderation, and oversees the system's overall integrity.

2. Clients - User who register to ^{post} project requirements, manage listings, receive applications and hire freelancers based on their proposals.
3. Freelancers - Individuals who register, create detailed profiles, browse available projects and submit tailored proposals for those that align with their expertise.

Objective :-

The primary objective of this project is to build a robust and user-friendly web-based freelancing platform that streamlines the process of connecting freelancers with potential clients. The key objectives include:

1. User Role Management :

Enable registration and secure login for three types of users : freelancers, clients and admin. Provide role-specific features and dashboards to ensure clarity and usability.

2. Project Posting and Listing :

Allow clients to post detailed requirements, including category and price. Provide options to edit, update or remove projects from their account.

3. Project Discovery :

Allow freelancers to view and filter available projects based on their skills and interests. Enable freelancers to apply with customer proposals and track their application status.

4. Admin Control Panel :

Allow admins to manage the platform by adding/editing project categories, managing user accounts and monitoring all posted content. Provide admin access to view platform analytics, reported content, and overall activity logs.

5. Category-Based Organization :

Implement an organized structure by dividing projects into categories, making it easier for freelancers to find relevant work.

supremo®

Problem Statement :-

In the current digital marketplace, there is a growing disconnect between businesses that require on-demand services and individuals who offer specialized freelance skills. There is a need for a specialized, simplified, affordable, and structured platform that allows:

Clients to easily post their project requirements, freelancers to find and apply for relevant opportunities, admin to manage and maintain platform operations effectively.

Scope of the project :-

The scope of ProConnect includes the design, development, testing and deployment of a fully functional freelancing web platform that serves three primary users: Freelancers, Clients and Admins.

Key features within the scope :

1. User Registration and Authentication for freelancers, clients and admin.
2. Client panel to create, manage and delete project listings.

3. Freelancer Panel to search projects, apply by submitting proposals, and manage applications.
4. Admin panel to manage users, categories and maintain platform moderation.
5. Category Management for organizing projects based on domains such as development, design, writing etc.
6. Database Integration to handle user data, project listings and proposals.
7. Responsive Frontend Design to ensure accessibility across devices.

Technologies Used :-

1. Frontend : HTML, CSS, JS
2. Backend : JAVA, SPRINGBOOT FRAMEWORK
3. Database : MySQL
4. Authentication : springBoot

Functional working of the website :-

1. User Roles :

- Admin
- Client
- Freelancer

2. Modules and their Functionalities :

A. Authentication Module

- User Registration (Freelancer / Client)
- Login / logout functionality
- password encryption

B. Freelancer Module

- Register and Create a profile with skills, portfolio etc.
- Browse available projects by category
- Apply for projects with a cover letter & proposals
- View applied projects and application status
- Manage profile settings

C. Client Module

- Register and Create a personal profile
- post new projects
- edit / delete posted projects
- View applications from freelancers
- Shortlist and contact freelancers
- Manage own project listing

D. Admin Module

- Admin login
- Add / Edit / Delete project categories
- View and manage all users (freelancers & clients)
- Monitor project listings and applications
- Manage reported users or spam content

E. Project Listing & Browsing Module

- Filter projects by category
- Search bar for keywords
- pagination or load-more functionality

F. Dashboard Module

- Freelancer Dashboard : project request, profile view etc
- Client Dashboard : application received, posted project, listing application etc.
- Admin Dashboard : Total users, active projects, categories, analytics etc.

Expected Outcome :-

1. Fully Functional Web-Based Freelancing platform.
2. Efficient project posting and Bidding system
3. Admin Panel for platform Management.
4. Organized project listing by category.
5. Database Integration and Management.

supremo®

Conclusion :-

The development of ProConnect makes a significant step forward addressing the growing need for an efficient, user-friendly, and accessible freelancing platform. This project successfully demonstrates the design and implementation of a web-based system that supports key functionalities such as user registration, project posting, category management, and role-based access control. By streamlining the process of hiring and applying for freelance projects, ProConnect not only enhances user productivity but also improves the overall experience for both freelancers and clients.

Week - 1

13 - 01 - 25

System Setup :-

1. Installation and System checking of V8 code with extensions of Java extension pack and SpringBoot extension pack .
2. Download and Install Java JDK 17 .
3. Download and Install Git and Setup Git for the project .
4. Download and Register in Github and Create a repository for the project .

supremo®

5. Install mysql and setup environment setting in mysql .
6. Create a SpringBoot project using Spring Initializer i.e start.spring.io .
7. Select the following :
Maven project , Java language ,
3.4.3 Springboot version , project
metadata - group : edu.oms ,
artifact : PoConnect , name : Po-Connect ,
package name : edu.oms.Po-Connect ,
jar packaging , 17 Java and
dependencies .
8. The following dependencies are added in the project :

- Spring Web : Build web , including RESTful , application using spring MVC .
Uses Apache Tomcat as the default embedded container .
- Thymeleaf : A modern server-side Java template engine for both web and stand-alone environments . Allows HTML to be correctly displayed in browsers and as static prototypes .

- Springboot DevTools : provides fast application restarts, LiveReload, and Configurations for enhanced development experience .
- MySQL Driver : MySQL JDBC driver .
- Spring Data JPA : persist data in SQL stores with Java persistence API using Spring Data and Hibernate .
- Validation (I/O) : Bean Validation with Hibernate Validator .
- Lombok (Developer Tools) : Java annotation library which helps to reduce boilerplate code .
 9. Then generate the spring project files .
- 10. Creating Zip folder on ProConnect and extraction of file .
- 11. Open and Run the program in VSCode with opening of the springboot application .
- 12. Test the extensions of Java and springboot are properly working .

Task Assigned :-

14 - 01 - 25

It is recommended to visit several websites while researching the project.

Exploring different sources will help you gain diverse perspective and a broader understanding. Each site may provide unique insights, which can contribute to a more comprehensive overview of the project's background and objectives.

15 15 - 01 - 25

- Creating the home page or landing page of a website as HTML file in the template folder with the CSS file in the static folder.
- The home page or landing page include header section, logo of the project, navigation section, contents in the body section with footer section.

16 - 01 - 25

- Accessing the project output through the URL (e.g - localhost : 8080 / client / home)
- Creating Controller for the landing page and learned how to connect pages using getmapping and requestmapping.

supremo®

17 - 01 - 25

- Adding the application properties as

server.port = 1211

Spring.datasource.url = jdbc:mysql://localhost

Spring.datasource.username = root

Spring.datasource.password = \${mysql.password}

Spring.jpa.hibernate.ddl-auto = update

- Set up mysql for the project.

18 - 01 - 25

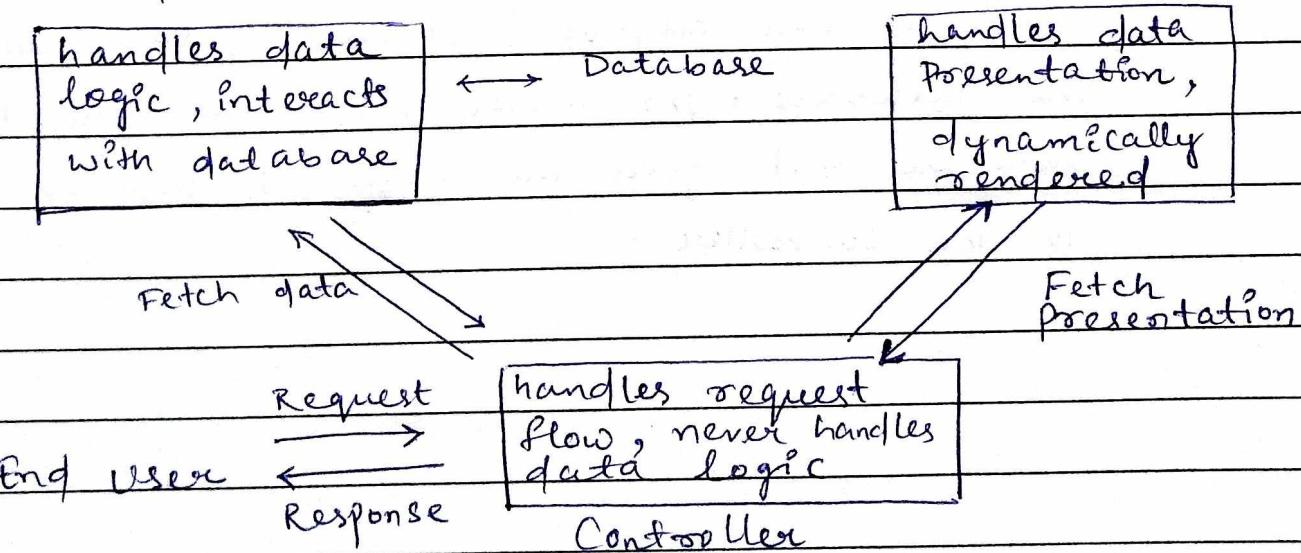
- Introducing PLP page which is product listing page and PDP page which is product detail page.

- Introducing 3-tier architecture and MVC pattern.

- MVC pattern is a Model-View-Controller

Model

View



Controller - It is the component that enables the interconnection between the views and model so it acts as an intermediary. It processes all the business logic and incoming requests, manipulates data using the Model component, and interact with the view to render the final output.

View - The view component is used for all the UI logic of the application. It generates a user interface for the user. Views are created by the data which collected by the model component through the controller.

Model - The model component corresponds to all the data-related logic that the user works with. It can add or retrieve data from the database. The model interacts with the database and gives the required data back to the controller.

19-01-25

- Introducing git setup and clone the git with github and learned to push and pull the code using git to push the code to the github repository.
- The push and pull of code were done by the collaborator of the project. ~~Samuels~~ 20-01-25

Week-2

20-01-25

- Created files under the corresponding folder (e.g - html file, css file, image file, controller etc).

21-01-25

- 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.

supremo®



AnyScanner

22-01-25

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

23-01-25

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

24-01-25

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

supremo®

25-01-25

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

26-01-25

- Worked on displaying the few details of the freelancer where the clients can easily make decision which freelancer is best for the project . ~~Supremo~~ 27-01-25

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

Week - 3

27-01-25

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

supremo®



AnyScanner

28 - 01 - 25

- 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, skills etc.

29 - 01 - 25

- Adjusting the pdp page of freelancer style for better design adding images of the freelancer.
- Including the features such as header, navigation, footer.

30 - 01 - 25

- Learned about agile model, scrum process, JDBC.
- Introducing the branching of the webpages of the website.

31 - 01 - 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.

1 ; 02 - 25

- Setting the rules of the github where
we can restrict the master branch so
that no other person can access the master
branch without reviewing.
- Learn how to merge code in the master
branch.
- Also learned pull and push using
pull request in the github where
we can compare between master branch
with other branches.
- After reviewing the code assignees can
submit the review then merge the
pull request.
- If there is any error in the code
during reviewing the person can also
leave a message for another review.

2-02-25

- Learn to create controller for each module such as

ClientController.java , FreelancerController.java ,
LoginController.java , CategoryController.java ,
ProjectController.java
ProjectRequestController.java
AdminDashboardController.java
ClientDashboardController.java
FreelancerDashboardController.java

~~Submitted~~
3.02.25

Week - 3

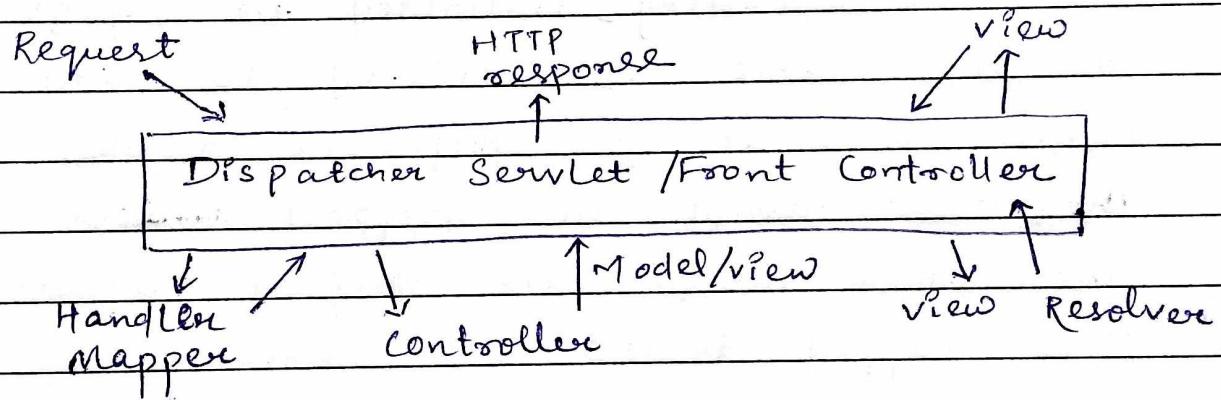
3-02-25

- Learned to create auditing class file where we need created_date , created_by , updated_date , updated_by for every java class file .
- Introduced AWS demand service .
- Also creating category table using auditing class and other features related to the category class file .

supremo®

4-02-25

- Creating database folder having `Script.sql` file and `data.sql` file where the query written in the script file for the database and data which are some sample for the database stored in the `data.sql` file.
- Also learned how the MVC pattern works as a flow diagram.



- Also learned different types of Java versions such as Java S.E (oracle), Java E.E (eclipse), Java M.E.

5-02-25

- 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 i.e. 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 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 - 02 - 25

- we learned how the mysql workbench work and how to execute sql query in the schema.
- First we need to make a 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 and @getter method.
- We can also set localDateTime for future reference.

7-02-25

- 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, linkedList.
- 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.

8-02-25

- 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 can be

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

9-02-25

- Creating table such as user table, Category table, project table, Freelancer table, payment table etc.
- Introducing getter and setter method, JDBC connection, JPA, ORM

Summary
10.02.25

Week - 4

10-2-25

- Introducing CI / CD workflow i.e. continuous Integration, continuous deployment.
- java object \rightarrow Table row data mapping
- ORM i.e. Object Relational mapping use hibernate (java), ibatis, Sequelizer (node.js).
- Difference between JPA and Hibernate

JPA :

1. It is described in javax.persistence package.
2. It describes the handling of relational data in java applications.
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 i.e @Autowired is annotation used for automatic dependency injection.
- Allowing spring to inject dependencies into our classes by type, eliminating the need for manual configuration.
- Field injection : Annotation of a field with @Autowired.
- Adding ~~categoryrepository~~ in client controller file where we add list of categories for listing purpose.

12-02-25

- Creating the repository class files are !

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

JPA repository -

- It 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(), for common database.
- Spring Boot embeds a servlet container in a cloud friendly way.

13 - 02 - 25

- Learned how to create or generate ticket
- GitHub:
 1. Create new issue
 2. Add title : Bind backend category data
 3. Description : Get data from DB and bind it on home screen to map the category.
 4. Apply labels to this issue : enhancement
- These steps are done for the category click should suggest related projects on plp page .
- Introducing Bitbucket, GitLab

14-02-25

- Addition of about us page , how it works page , Our Solution page , Contact us page .
- Changes made in CSS of all web pages .

supremo®



AnyScanner

15-02-25

- Worked on mysql workbench where we can export data between collaborators.
- Steps are :

Server → option → select database
→ Export command → start export

- For successful exporting data using a application i.e BeekeeperStudio .

16-02-25

- Creating Constant files such as
 - i) CategoryStatus . java
 - ii) PaymentStatus . java
 - iii) ProjectRequestStatus . java
 - iv) projectStatus . java
 - v) ServiceRequestStatus . java
 - vi) UserType

Surendra

17-02-25

Week-5

17-02-25

- Working on payment page , registration , sign in page .
- Also worked on category pdp page
- Working on category table .

18-02-25

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

19-02-25

- worked on pdp page of the category page .
- Fixing the issue in the category page connection .

20-02-25

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

supremo®



AnyScanner

21 - 02 - 25

- Learned post mapping concept , form : get and post method where get - show user name and password and post will not show .
- Used form method post in client sign-up page .
- Created freelancer dashboard .

22 - 02 - 25

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

23 - 02 - 25

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

Smiti
24.02.25

WEEK-6

24-02-21

- Learned thymeleaf 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> findByCategoryName
ContainingIgnoreCase (String categoryName)`

25-02-21

- 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/jpeg" />`
- Also connected using postmapping ,
`@requestparam , @ModelAttribute ,
CommandLineRunner` .

supremo®



AnyScanner

26-02-25

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

@ Override

public void run (String... args) throws exception

```
{ File file = new File ( pathname : "upload-  
images" );
```

```
if ( file . exists () )
```

```
{ file . mkdirs ();
```

```
}
```

```
}
```

27-02-25

- Worked on category page of admin dashboard .
- Introducing UI/UX concept ^{and} where it implemented .

28-02-25

- Addition of form in category page of admin dashboard :

01-03-25

- worked on image uploading in admin dashboard .
- Implementation of UVID concept in the image uploading .

02-03-25

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

Amritpal
03-03-25

supremo®

Week-7

3-03-25

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

4-03-25

- Worked on adding data on the project page of the client dashboard.
- Addition of search query
 - @ Controller
 - @ Request mapping
 - @ Get mapping ("search")
String search (@ Request param String search)

5-03-25

- Worked on backend end 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're who they claim to be.

It involves checking credentials which include username, password.

- Authorization :-

It is the method of figuring out and granting permissions to a demonstrated user or system, specifying what assets they can access and what actions they're allowed to carry out.

06-03-25

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

@ Bean

```
passwordEncoder encoder()  
{  
    return BcryptPasswordEncoder()  
}
```

- Addition of encoder in the login
- Addition of UserDetailsService
by using UserRepository.

07-03-25

- worked on search bar and logout option added.
- Learning addition of Thymeleaf security dependency.
- @ AllArgsConstructor, @NoArgsConstructor
- Difference among warehousing, marketplace supplier, dropshipping
- Addition of UserService.java

8-03-25

- worked on addition of pop page and addition of pages in other dashboard.
- Addition of earning page, payment page changes in its content.

9-03-25

- worked on linking my project and project description page.
- worked on profile page in client dashboard.

Supremo
10.03.25

supremo®

Week - 8

10-03-25

- Working on adding data in table and payment page .
- My project page and create another sign-in page one for client i.e user and another for freelancer .

11-03-25

- worked on sign-up page
- working 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 and client dashboard .
- Added service request page .

13 - 03 - 25

- Working on user management page and freelancer management .

14 - 03 - 25

- worked on service request page and project request page in the client dashboard .

15 - 03 - 25

- worked on service request page and project completion Request page in freelancer dashboard .

16 - 03 - 25

- Addition of profile page in freelancer dashboard .

~~Supremo~~
17 - 03 - 25

supremo®

Week - 9

17-03-25

- Addition of status , and action of the freelancer dashboard .
- Addition of the table in the project request .

18-03-25

- Created `OfferRepository` under the `repository` folder to extend `TPA repository` .

19-03-25

- Writing code for the edit button for the Category and project page .

20-03-25

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

supremo®

21-03-25

- Worked on search button or bar
Where we trying to search for
categories, projects, and freelancers
profile.

22-03-25

- Working on spring security for
the sign-in page.

Supremo
22.03.25

supremo®



AnyScanner