



TRƯỜNG ĐẠI HỌC FPT

MINISTRY OF EDUCATION AND TRAINING

FPT UNIVERSITY

Capstone Project Document

[WDCRBP]

SP25SE026_GSP48	
Group Members	Trần Việt Phúc - SE171429 Võ Trọng Luân - SE171436 Nguyễn Đặng Phương Nam - SE171442 Lê Nguyên An - SE171475
Supervisor	Trần Ngọc Như Quỳnh
Capstone Project code	SP25SE026

- Ho Chi Minh, 04/2025 -

Table of Contents

Acknowledgement	7
Definition and Acronyms.....	8
I. Project Introduction	8
1. Overview.....	8
1.1 Project Information	8
1.2 Project Team	8
2. Product Background.....	9
3. Existing Systems.....	10
3.1 Etsy (www.etsy.com).....	10
3.2 Houzz (www.houzz.com)	10
4. Business Opportunity.....	11
5. Software Product Vision.....	11
6. Project Scope & Limitations	11
6.1 Major Features	11
6.2 Limitations & Exclusions.....	13
II. Project Management Plan.....	14
1. Overview.....	14
1.1 Scope & Estimation	14
1.2 Project Objectives	15
1.3 Project Risks	15
2. Management Approach.....	15
2.1 Project Process	16
2.2 Quality Management.....	17
2.3 Training Plan.....	17
3. Project Deliverables	17
4. Responsibility Assignments.....	18
5. Project Communications.....	18
6. Configuration Management	19
6.1 Document Management	19
6.2 Source Code Management	19
6.3 Tools & Infrastructures	19
III. Software Requirement Specification	20
1. Product Overview	20
2. User Requirements.....	20
2.1 Actors.....	20
2.2 Use cases	21

3. Functional Requirements	101
3.1 System Functional Overview	101
3.2 Entity Relationship Diagram.....	105
4. Non-Functional Requirements	138
4.1 External Interfaces	138
4.2 Quality Attributes.....	138
5. Requirement Appendix	138
5.1 Business Rules	138
5.2 Application Message List.....	141
IV. Software Design Description	142
1. System Design	142
1.1 System Architecture.....	142
1.2 Package Diagram	143
2. Database Design.....	146
3. Detailed Design.....	163
3.1 Woodworker registration feature	163
3.2 Customization service booking feature.....	166
3.3 Personalization service booking feature.....	170
3.4 Guarantee service booking feature.....	172
V. Software Testing Documentation.....	175
1. Overall Description.....	175
1.1 Test Scope.....	175
1.2 Test Model	175
1.3 Testing Levels.....	175
2. Test Plan.....	175
2.1 Test Stages	175
2.2 Resources	175
2.3 Test Milestones	175
3. Test Cases and Test Reports	175
VI. Release Package & User Guides.....	176
1. Deliverable Package.....	176
2. Installation Guides	176
2.1 System Requirements.....	176
2.2 Setup Files.....	177
2.3 Installation Instruction	177
3. User Manual.....	181
3.1 Overview.....	181

3.2 Web application	181
3.3 Mobile application	203

List of Tables

Table 1 - Definition and Acronyms	8
Table 2 - Supervisor.....	8
Table 3 - Team member.....	9
Table 4 - Scope & Estimation.....	15
Table 5 - Project Objectives.....	15
Table 6 - Project Risks.....	15
Table 7 - Training Plan	17
Table 8 - Project Deliverables.....	18
Table 9 - Responsibility Assignments	18
Table 10 - Project Communications.....	19
Table 11 - Tools & Infrastructures.....	19
Table 12 - Actors	21
Table 13 - ERD dictionary	138
Table 14 - Business Rules.....	141
Table 15 - Application Message List	141
Table 16 - Database data dictionary.....	163
Table 17 - Test Stages.....	175
Table 18 - Test Resources.....	175
Table 19 - Test Milestones.....	175
Table 20 - Test Cases and Test Reports.....	175
Table 21 - Deliverable Package	176
Table 22 - System Requirements Web Application.....	176
Table 23 - System Requirements Mobile Application.....	176
Table 24 - System Requirements Web API	176

List of Figures

Figure 1 - Project Process	16
Figure 2 - Product backlog.....	17
Figure 3 - Context Diagram	20
Figure 4 - Use case diagram.....	21
Figure 5 - Customer Web Application / Mobile Application Screen Flow	101
Figure 6 - Woodworker Web Application / Mobile Application Screen Flow.....	102
Figure 7 - Staff Web Application Screen Flow.....	103
Figure 8 - Moderator Web Application Screen Flow.....	103
Figure 9 - Admin Web Application Screen Flow	104
Figure 10 - Entity Relationship Diagram.....	105
Figure 11 - System Architecture	142
Figure 12 - Web API package diagram.....	143
Figure 13 - Web application package diagram	144
Figure 14 - Mobile application package diagram	145
Figure 15 - Database design.....	146
Figure 16 - Woodworker registration feature class diagram.....	163
Figure 17 - Woodworker registration activity diagram.....	164
Figure 18 - Register sequence diagram.....	165
Figure 19 - Approval sequence diagram	166
Figure 20 - Customization service booking feature class diagram	166
Figure 21 - Customization service booking feature activity diagram	167
Figure 22 - Book customization service sequence diagram.....	168
Figure 23 - Update consultant appointment for customization service sequence diagram	169
Figure 24 - Customization order state machine diagram	169
Figure 25 - Personalization service booking feature class diagram	170
Figure 26 - Personalization service booking feature activity diagram.....	171
Figure 27 - Personalization service booking feature sequence diagram	171
Figure 28 - Personalization order state machine diagram	172
Figure 29 - Guarantee service booking feature class diagram	172
Figure 30 - Guarantee service booking feature activity diagram	173
Figure 31 - Request guarantee order sequence diagram	173
Figure 32 - Accept guarantee order sequence diagram.....	174
Figure 33 - Guarantee order state machine diagram	174

Acknowledgement

We would like to express our deepest gratitude to our supervisor, Ms. Trần Ngọc Nhu Quỳnh, for her invaluable support and guidance throughout the development of this project. Her expertise and insightful feedback have played a crucial role in refining our work and pushing us toward improvement.

Additionally, we extend our sincere appreciation to:

- The lecturers at FPT University, whose knowledge and mentorship have shaped our academic journey and equipped us with the necessary skills to undertake this project.
- Mr. Nguyễn Ngọc Lâm, Mr. Nguyễn Minh Sang, and other faculty members who provided constructive criticism and valuable feedback, helping us identify and improve areas that needed further refinement.
- Our families and friends for their unwavering encouragement and support, standing by us through every challenge we encountered.
- A special thanks once again to Ms. Trần Ngọc Nhu Quỳnh for her dedication and continuous assistance in various aspects of the project, from technical expertise to business insights and documentation.

We acknowledge that, given our limited experience and the complexity of certain technologies, mistakes were inevitable. We sincerely hope that any shortcomings will be met with understanding. Thank you to everyone who has supported us throughout this journey.

Definition and Acronyms

Acronym	Definition
PWM	Psychology website
AWS	Amazon Web Services
BA	Business Analysis
BR	Business Rule
ERD	Entity Relationship Diagram
GUI	Graphical User Interface
PM	Project Manager
SDD	Software Design Description
SPMP	Software Project Management Plan
SRS	Software Requirement Specification
UAT	User Acceptance Test
UC	Use Case
API	Application Program Interface

Table 1 - Definition and Acronyms

I. Project Introduction

1. Overview

1.1 Project Information

- Project name: Woodwork Design, Crafting & Repair Booking Platform
- Project code: SP25SE026
- Group name: SP25SE026_GSP48
- Software type: Web application, Mobile application

1.2 Project Team

1.2.1 Supervisor

Full Name	Role	Email	Mobile
Trần Ngọc Như Quỳnh	Lecturer	quynhtnn4@fpt.edu.vn	0779989791

Table 2 - Supervisor

1.2.2 Team member

Full Name	Role	Email	Mobile
Võ Trọng Luân	Leader	luanvtse171436@fpt.edu.vn	0971781359
Trần Việt Phúc	Member	phucTVSE171429@fpt.edu.vn	0944595483
Lê Nguyên An	Member	anlnse171475@fpt.edu.vn	0902846869
Nguyễn Đặng Phương Nam	Member	namndpse171442@fpt.edu.vn	0906657671

Table 3 - Team member

2. Product Background

Customers looking for high-quality, custom woodwork often struggle to find skilled and reliable woodworkers. Many face challenges such as unclear pricing, limited customization options, and difficulty in assessing service quality. Communication between customers and woodworkers is often inefficient, leading to delays and unmet expectations.

Woodworkers also face difficulties in reaching potential customers, managing bookings, and showcasing their expertise. Without a centralized platform, they rely on word-of-mouth or social media, which limits their opportunities for growth.

To solve these issues, the Woodwork Design, Crafting & Repair Booking Platform was proposed. It provides a trusted online marketplace where customers can easily find, review, and book skilled woodworkers. The platform ensures transparent pricing, customization options, and direct communication between customers and woodworkers, creating a more efficient and reliable experience for both parties.

3. Existing Systems

Currently, there are a few platforms and marketplaces that offer woodworking services or customized furniture. However, these systems often have limitations in terms of service coverage, transparency, and communication between customers and woodworkers. Below are two examples of existing systems that serve similar purposes:

3.1 Etsy (www.etsy.com)

Description: Etsy is an e-commerce platform that allows artisans, including woodworkers, to sell handcrafted products. Users can browse, purchase, and communicate with sellers.

Features:

- Listing of handcrafted wood products
- Communication through messaging
- Payment integration

Pros:

- Large marketplace for handmade items
- Customer reviews and ratings available

Cons:

- Does not focus on service booking
- Lack of direct customization options for projects
- Do not focus on the field of wooden furniture

3.2 Houzz (www.houzz.com)

Description: Houzz is an online platform that connects homeowners with professionals, including woodworkers and furniture designers.

Features:

- Professional portfolio display

- Service booking for home improvement
- Customer reviews and ratings

Pros:

- Verified professionals
- Extensive project galleries

Cons:

- High competition among service providers
- Do not focus on the field of wooden furniture

4. Business Opportunity

The woodworking industry is growing as more people seek custom, high-quality wooden products for their homes and businesses. However, the current process of finding reliable woodworkers remains inefficient. Many skilled woodworkers lack a structured way to showcase their expertise, while customers struggle with unclear pricing, poor service quality assessment, and limited customization options.

The Woodwork Design, Crafting & Repair Booking Platform aims to address these gaps by:

- Providing a centralized marketplace for customers to discover skilled woodworkers.
- Offering customization options to allow users to specify their exact needs.
- Implementing transparent pricing and a rating system to ensure service quality.
- Enabling direct communication between customers and woodworkers to improve project clarity.

This platform has the potential to become a leading solution in the woodworking service industry by ensuring accessibility, reliability, and convenience for both customers and professionals.

5. Software Product Vision

For individuals and businesses in need of high-quality woodworking services, the Woodwork Design, Crafting & Repair Booking Platform is an online marketplace that facilitates seamless discovery, booking, and communication with skilled woodworkers. Unlike existing platforms, our solution provides customization, transparent pricing, and a verified service provider network, ensuring an efficient and trustworthy experience.

6. Project Scope & Limitations

6.1 Major Features

6.1.1 Guest use mobile application / web application

FE-01: Search, filter, view list, view detail woodworker.

FE-02: Search, filter, view list, view detail product offered by woodworker.

FE-03: Search, filter, view list, view detail design idea offered by woodworker.

FE-04: Register.

6.1.2 Customer use mobile application / web application

FE-05: Order product.

FE-06: Book customization service.

FE-07: Book personalization service.

FE-08: Book guarantee service.

FE-09: Manage service order: view list, view detail.

FE-10: Manage guarantee order: view list, view detail.

FE-11: Manage complaint: create complaint, view list, view detail.

FE-12: Mange address: add address, view list, update address.

FE-13: Manage wallet: view transaction history, view transaction detail, deposit money, withdraw money.

FE-14: Manage account: create account, view profile, update profile, change password, reset password.

6.1.3 Woodworker use mobile application / web application

FE-15: Manage service: view list, update service.

FE-16: Manage product: create product, view product list, view product detail, update product, delete product.

FE-17: Manage post: create post, view post list, view post detail, update post, delete post.

FE-18: Manage review: view review list, view review deatail, respond to review.

FE-19: Manage design idea: create design idea, view design idea list, view design idea detail, update design idea, delete design idea.

FE-20: Manage service order: view order list, view order detail.

FE-21: Manage guarantee order: view order list, view order detail.

FE-22: Change profile visibility.

FE-23: Upgrade service pack.

FE-24: Manage wallet: view transaction history, view transaction detail, deposit money, withdraw money, link bank account.

FE-25: Manage customer complaint: view complaint list, view complaint detail, respond to complaint.

FE-26: Manage account: create account, view profile, update profile, reset password.

6.1.4 Moderator use admin web application

FE-27: View woodworker performance: view list, view detail.

FE-28: Send performance email to woodworker.

6.1.5 Staff use admin web application

FE-29: Manage review: view review list, approve/reject review.

FE-30: Manage woodworker's reponse: view response list, view response detail, review woodworker's response.

FE-31: Manage complaint: view complaint list, view complaint detail, handle complaint.

6.1.6 Admin use admin web application

FE-32: Manage configuration: view configuration list, update configuration.

FE-33: Manage woodworker registration: view registration list, view registration detail, review registration.

FE-34: Manage payment: view transaction history, view transaction detail, view transaction & revenue chart.

6.2 Limitations & Exclusions

LI-1: Physical workshop locations: The platform will not provide physical workshops or showrooms; it only connects customers and woodworkers.

LI-2: Automated design suggestions: No AI-powered design recommendations in the initial version.

LI-3: Delivery & logistics: The platform does not handle transportation or delivery of woodwork products.

LI-4: International services: Initially limited to a specific geographic region.

II. Project Management Plan

1. Overview

1.1 Scope & Estimation

#	WBS Item	Complexity	Est. Effort (man-days)
1	Initiating		
1.1	Collect requirement	Medium	14
2	Planning		
2.1	Create plan document	Complex	10
3	Executing		
3.1	Analysis		
3.1.1	Analysis requirement	Complex	12
3.2	Design and prototyping		
3.2.1	Design customer web/mobile application	Complex	20
3.2.2	Design customer web/mobile application	Complex	20
3.2.3	Design platform's staff web application	Complex	15
3.3	Implementation		
3.3.1	Implement base feature		
3.3.1.1	Login and logout	Simple	2
3.3.1.2	Update profile	Simple	2
3.3.1.3	Register	Simple	2
3.3.1.4	Change and reset password	Simple	2
3.3.2	Implement woodworker feature		
3.3.2.1	Manage service	Medium	4
3.3.2.2	Manage product	Medium	2
3.3.2.3	Manage design idea	Complex	8
3.3.2.4	Manage post	Medium	2
3.3.2.5	Manage service order	Complex	20
3.3.2.6	Manage guarantee order	Complex	20
3.3.2.7	Manage wallet	Medium	10
3.3.2.8	Manage customer complaint	Medium	5
3.3.2.9	Manage review	Medium	2
3.3.3	Implement customer feature		
3.3.3.1	Search, filter, view list/detail woodworker	Medium	3
3.3.3.2	Search, filter, view list/detail product	Medium	3
3.3.3.3	Search, filter, view list/detail design idea	Medium	3
3.3.3.4	Book and manage customization service	Complex	15
3.3.3.5	Book and manage personalization service	Complex	15
3.3.3.6	Review and rating order	Medium	2
3.3.3.7	Manage wallet	Medium	10
3.3.3.8	Manage complaint	Medium	10
3.3.4	Implement moderator feature		

3.3.4.1	Manage woodworker performance	Medium	6
3.3.4.2	Send performance email	Medium	3
3.3.5	Implement staff feature		
3.3.5.1	Manage customer's review	Medium	3
3.3.5.2	Manage woodworker's response	Medium	3
3.3.5.3	Manage complaint	Complex	10
3.3.6	Implement admin feature		
3.3.6.1	Manage woodworker registration	Medium	10
3.3.6.2	Manage configuration	Medium	10
3.4	Testing		
3.4.1	Unit testing	Complex	12
3.4.2	Integration test	Complex	12
3.4.3	System test	Complex	12
4	Monitoring and controlling		
4.1	Control the process	Complex	12
4.2	Track performance and quality	Complex	12
5	Closing		
5.1	Report	Simple	20

Total Estimated Effort (man-days) **358**

Table 4 - Scope & Estimation

1.2 Project Objectives

#	Testing Stage	No. of Defects	% of Defect	Notes
1	Reviewing	20	27,6%	
2	System Test	40	72,4%	
3	User Acceptance Test	X	X	
Total		60	100%	

Table 5 - Project Objectives

1.3 Project Risks

#	Risk Description	Impact	Possibility	Response Plans
1	Technology risk: cannot find the suitable solution for technique problems, the software cannot fulfil the expectation.	High	Medium	Do more research on official resources, discuss with mentor.
2	Requirement changes: requirements unexpected Changes.	Critical	Medium	Chose another way of solution to respond to those changes.

Table 6 - Project Risks

2. Management Approach

2.1 Project Process

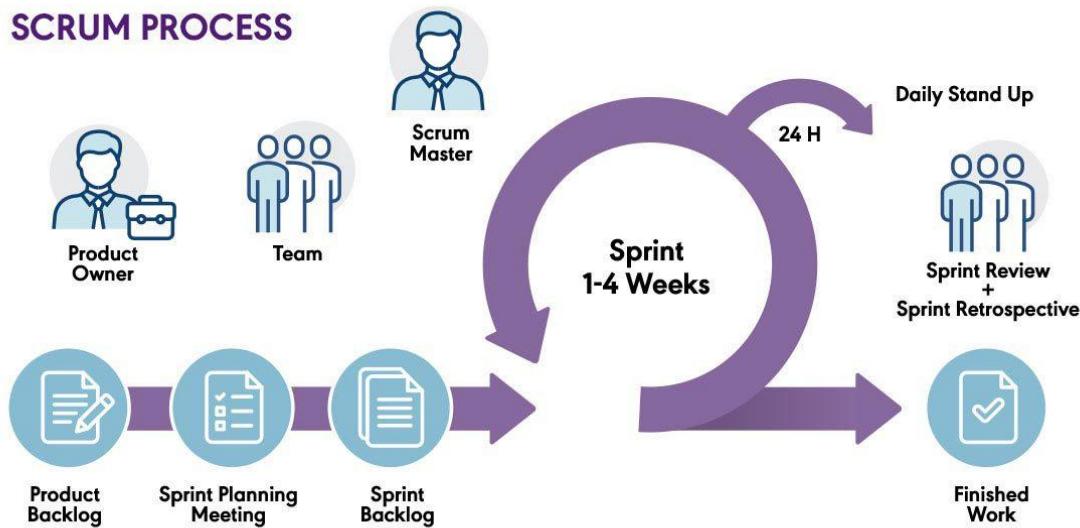


Figure 1 - Project Process

References: [What is Scrum? | The Agile Journey: A Scrum overview](#)

Our project, adopted the Scrum model, an Agile framework for software development, for the following reasons:

- We worked with a small team of 4 members, making it unnecessary to follow every stage of a traditional methodology like Waterfall. Instead, Scrum allowed us to iterate quickly and focus on delivering value continuously.
- By organizing work into short sprints, we were able to easily analyze the next priorities and refactor whenever necessary, ensuring flexibility throughout the development cycle.
- The limited time frame of each sprint created a healthy pressure, motivating the team to complete tasks efficiently and maintain momentum across the project.

Product backlog

Figure 2 - Product backlog

2.2 Quality Management

To maintain code and project quality:

- Conduct regular code reviews.
- Enforce consistent coding standards.
- Feedback loop: Demos and user feedback sessions every sprint.

2.3 Training Plan

Training Area	Participants	When, Duration	Waiver Criteria
Git & GitHub Workflow	All Members	Week 1 only	Mandatory
Jira + Agile Practice	All Members	Week 1 only	Mandatory
ReactJS Essentials	Võ Trọng Luân	Week 1 – Week 2	Mandatory
React Native Basics	Võ Trọng Luân Lê Nguyên An Nguyễn Đăng Phương	Week 1 – Week 2	Mandatory
Spring Boot Core	Trần Việt Phúc Nguyễn Đăng Phương Nam	Week 1 – Week 2	Mandatory

Table 7 - Training Plan

3. Project Deliverables

#	Deliverable	Due Date	Notes
1	User Requirements Document (URD)	22/02/2025	Defines user needs and expectations.
2	Software Requirements Specification (SRS)	22/02/2025	Outlines detailed functional and non-functional requirements.

3	System Architecture Design	22/02/2025	Includes high-level system design, database schema, and component interactions.
4	Web Application	05/04/2025	A web-based platform for guests, customers, woodworkers, and administrators with full functionality for booking services, reviews, account management, etc.
5	Mobile Application	05/04/2025	A user-friendly mobile app optimized for seamless user experience with features like the web application.
6	Backend System (RESTful API)	05/04/2025	Backend system ensuring authentication, service management, chat support, cost calculations, and payment integration.
7	Database System	22/02/2025	Data management system using MySQL/SQL Server with key tables such as Users, Bookings, Services, Payments, Reviews, etc.
8	Deployment & Hosting Deliverables	05/04/2025	Includes source code, server configuration, and deployment documentation on Railway.
9	Final Report & Presentation	28/04/2025	A final report summarizing the development process, achieved results, challenges faced, along with a presentation and product demo video

Table 8 - Project Deliverables

4. Responsibility Assignments

D~Do; R~Review; S~Support; I~Informed; <blank>- Omitted

Responsibility	luanvtse171436	phuctvse171429	anlnse171475	namndpse171442
Project Planning & Tracking	D	D	D	D
Prepare Project Introduction Document	D	D	D	D
Prepare SRS Document (Overview Part)	D	D	D	D
Prepare SRS Document (User Requirements)	D	D	D	D
Develop System Architecture	D	D	D	D
Implement Backend Components	R	D	R	D
Implement Frontend Components	D	R	R	R
Implement App Components	S	R	D	S

Table 9 - Responsibility Assignments

5. Project Communications

Communication Item	Who/ Target	Purpose	When, Frequency	Type, Tool, Method(s)

Texting	Supervisor + Team member	<ul style="list-style-type: none"> • Confirm Meeting time • Report Working progress • Internal conversations 	Usually	Zalo
Meeting offline	Supervisor + Team member	<ul style="list-style-type: none"> • Progress report • Report and handle problem • Review task 	Weekly	
Tracking progress	Team member	<ul style="list-style-type: none"> • Check personal tasks • Tracking progress of project on Jira 	Daily	Jira

Table 10 - Project Communications

6. Configuration Management

6.1 Document Management

- Use OneDrive to save document and for real-time editing on Microsoft Word.
- Use Google Drive to save meeting report, meeting record, diagram, image, design, document to each category folder

6.2 Source Code Management

- GitHub server to store source code.
- Follow the rule of team when using Git:
 - Commit and note the changes details often.
 - Review carefully before push into develop branch.
 - Fetch, review and resolve conflicts when merging usually

6.3 Tools & Infrastructures

Category	Tools / Infrastructure
Technology	React (Web), React Native (Mobile), Spring Boot (Back-end)
Database	MySQL
IDEs/Editors	Visual Studio Code, IntelliJ
Diagramming	Visual Paradigm, DrawIO, Mermaid Chart
Documentation	Ms Office, Google Docs/Sheets/Slides
Version Control	GitHub (Source Codes), Google Drive (Documents)
Deployment server	Railway, Vercel
Project management	Jira (Schedule), GitHub (Tasks, Defects)

Table 11 - Tools & Infrastructures

III. Software Requirement Specification

1. Product Overview

WDCRP Platform is a software system that connects customers with woodworkers, providing ultra-fast connection solutions to save time for both parties. The context diagram below illustrates the external entities (customers, woodworkers, payment gateway, etc.) and the corresponding system interfaces.

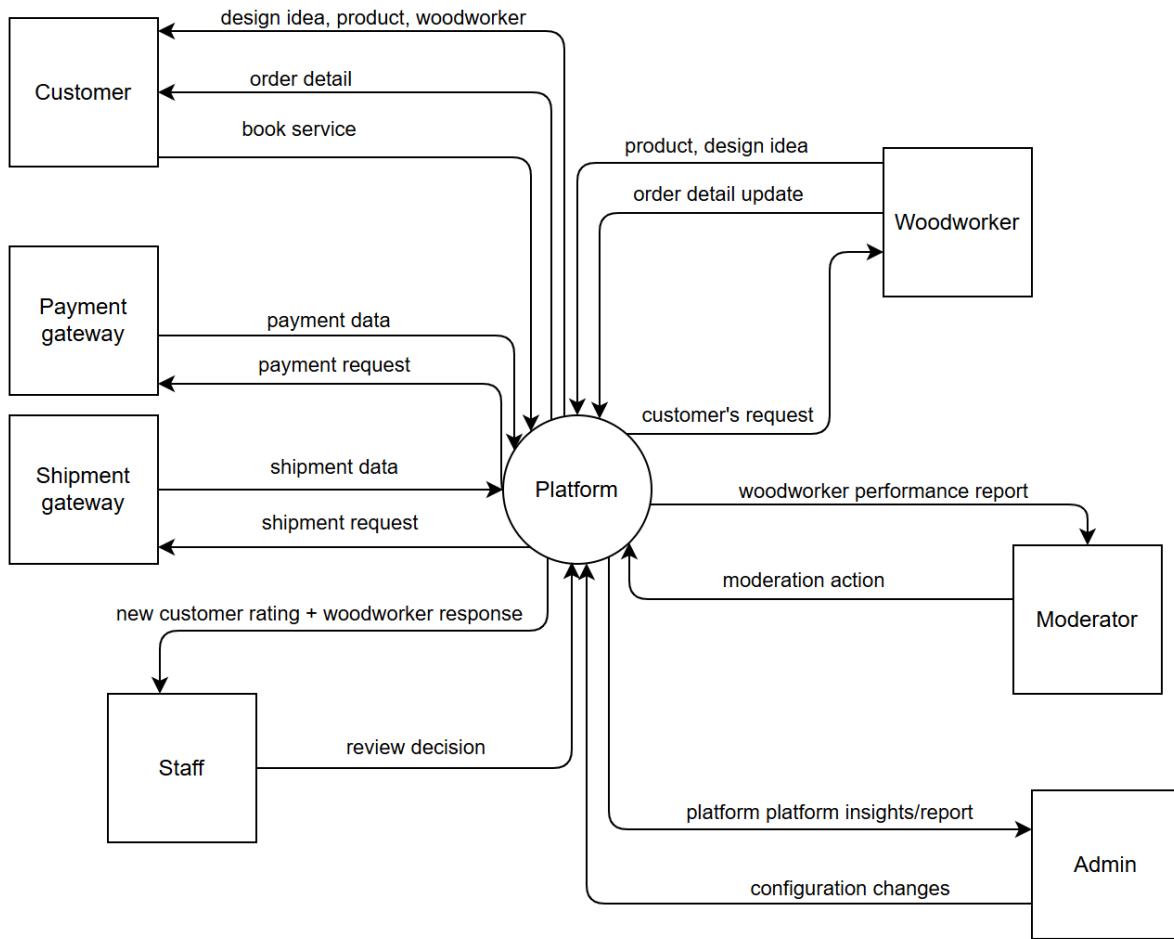


Figure 3 - Context Diagram

2. User Requirements

2.1 Actors

#	Actor	Description
1	Admin	The admin who configures platform settings and manages woodworker registration.
2	Moderator	The moderator who reviews woodworker's performance and sends emails.
3	Staff	The staff who reviews customer reviews, woodworker's response, and complaints
4	Woodworker	The woodworker who receives orders, crafts products according to design specs, and updates order detail.
5	Customer	The customer who books services and provides ratings.
6	Guest	The guest who explores woodworker portfolios without logging in.

7	Authenticated user	User who is logged in.
8	Unauthenticated user	User who is not logged in.

Table 12 - Actors

2.2 Use cases

2.2.1 Diagram

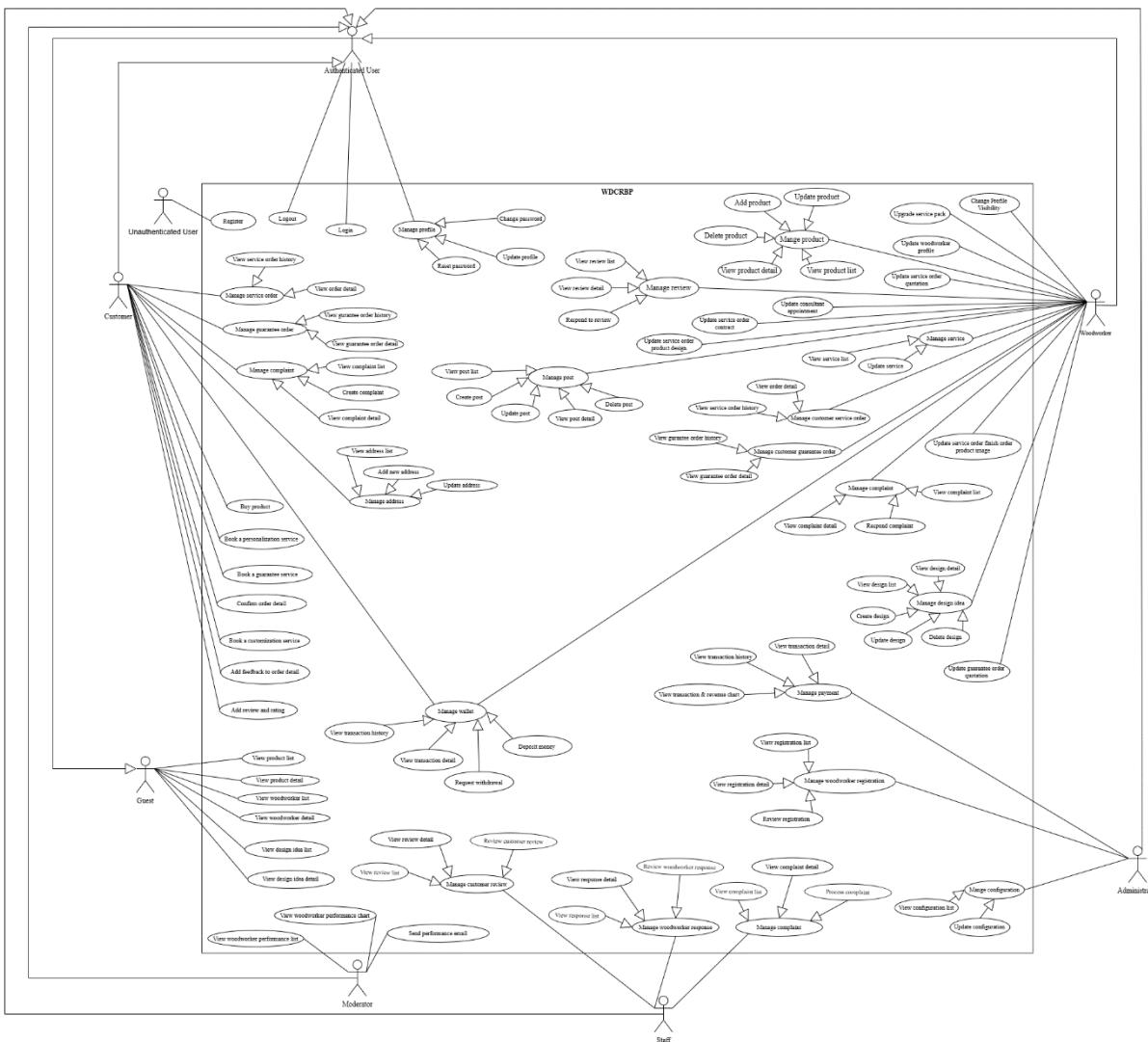


Figure 4 - Use case diagram

2.2.2 Use case list with description

UC-01: Register

Use Case ID and Name:	UC-01 - Register		
Created By:	An	Date Created:	23/04/2025
Primary Actor:	Unauthenticated user	Secondary Actors:	N/A

Trigger:	The user wants to create an account to access the system's features.
Description:	This use case describes the registration process for new users. Users can choose to register either as a customer or as a woodworker. Both flows require email verification via OTP. If registering as a woodworker, the user must additionally submit workshop information, which is subject to admin approval.
Preconditions:	PRE-1. User is not logged in. PRE-2. User has a valid email address and phone number. PRE-3. User has access to the Internet.
Post-conditions:	POST-1. For Customer: The user successfully registers and can log in to use the platform. POST-2. For Woodworker: The user successfully registers, submits workshop details, and awaits admin approval before accessing woodworker features.
Normal Flow:	<p>1.0 Customer Registration</p> <ol style="list-style-type: none"> 1. User clicks "Register". 2. System displays customer registration form. 3. User fills in name, email, password, confirm password, and phone number. 4. User clicks "Register". 5. System sends OTP to the provided email. 6. User enters OTP to verify the account. 7. System confirms registration success. 8. User can now log in and use customer features. <p>1.1 Woodworker Registration</p> <ol style="list-style-type: none"> 1. User clicks "Register as Woodworker" directly. 2. System displays a full registration form 3. User completes the form and clicks "Submit". 4. System creates an account and stores the workshop info. 5. System sets account status to Pending Approval. 6. Admin will later review and approve/reject.
Alternative Flows:	N/A
Exceptions:	N/A
Priority:	High
Frequency of Use:	Usually
Business Rules:	BR-01, BR-02
Other Information:	N/A
Assumptions:	User can check their email to complete OTP verification (customer only).

UC-02: Login

Use Case ID and Name:	UC-02 – Login		
Created By:	An	Date Created:	09/04/2025
Primary Actor:	Authenticated user	Secondary Actors:	
Trigger:	User chooses to sign in.		
Description:	User logs in using either an email + password combination or a one-time OTP sent to their email.		
Preconditions:	PRE-1: User has previously registered and has a verified email address.		
Post-conditions:	POST-1: User is authenticated and redirected to their dashboard.		
Normal Flow:	1.0 Sign in <ol style="list-style-type: none"> 1. User navigates to the login page. 2. System presents two options: Password login or Email OTP login. 3. User selects their preferred method. 4. User provides credentials (email + password or requests OTP and enters it). 5. System validates credentials or OTP. 6. System establishes an authenticated session and redirects the user. 		
Alternative Flows:	1.1 Use password login <ol style="list-style-type: none"> 1. Follows steps 4–6 using email + password. 1.2 Use email OTP login <ol style="list-style-type: none"> 1. At step 4, user requests an OTP; system sends it, user enters it; then continue with step 5. 		
Exceptions:	1.0.E1 Invalid credentials or OTP <ol style="list-style-type: none"> 1. At step 5, if authentication fails, system shows “Thông tin không hợp lệ.” 2. User may retry or choose the other login method. 		
Priority:	High		
Frequency of Use:	Usually		
Business Rules:	BR-17, BR-18		
Other Information:	N/A		
Assumptions:	Email delivery is timely and reliable.		

UC-03: Logout

Use Case ID and Name:	UC-03 – Logout		
Created By:		Date Created:	09/04/2025

Primary Actor:	Authenticate d user	Secondary Actors:	
Trigger:	User clicks “Đăng xuất” (Logout) in the application header or menu.		
Description:	The user ends their session, causing the system to invalidate their authentication and return them to the public-facing area.		
Preconditions:	PRE-1: User is currently logged in. PRE-2: A valid session/token exists on client and server.		
Post-conditions:	POST-1: The user’s session is terminated and any authentication tokens are revoked. POST-2: User is redirected to the home or login page.		
Normal Flow:	<p>1.0 Logout</p> <ol style="list-style-type: none"> 1. User clicks “Đăng xuất”. 2. System invalidates the user’s session on the server. 3. System clears authentication tokens/cookies in the client. 4. System redirects the user to the login or homepage. 		
Alternative Flows:	<p>1.1 Immediate Redirect</p> <p>If the application uses single-page routing, step 4 occurs without a full page reload.</p>		
Exceptions:	<p>1.0.E1 Session invalidation failure</p> <ol style="list-style-type: none"> 1. If the server cannot terminate the session, system displays “Không thể đăng xuất. Vui lòng thử lại.” and remains on the current page. 		
Priority:	High		
Frequency of Use:	Frequent		
Business Rules:	BR-19, BR20, BR21		
Other Information:	N/A		
Assumptions:	Network connection is available to notify the server.		

UC-04: Update profile

Use Case ID and Name:	UC-04 – Update profile		
Created By:	An	Date Created:	09/04/2025
Primary Actor:	Authenticated user	Secondary Actors:	
Trigger:	User selects “Cập nhật hồ sơ” (Update Profile) from their account menu.		

Description:	An authenticated user edits and saves changes to their personal details (e.g. full name, email, phone, address, avatar).
Preconditions:	<p>PRE-1: User is logged in</p> <p>PRE-2: A profile record already exists for the user</p> <p>PRE-3: Profile service is available</p>
Post-conditions:	<p>POST-1: The user's updated profile information is saved</p> <p>POST-2: The system displays a confirmation message and shows the new data</p>
Normal Flow:	<p>1.0 Update profile</p> <ol style="list-style-type: none"> 1. User clicks “Cập nhật hồ sơ” in their account menu. 2. System displays the profile form populated with current data. 3. User edits one or more fields and clicks Lưu (Save). 4. System validates the inputs, saves the changes, and shows “Cập nhật thành công.”
Alternative Flows:	<p>1.1 Cancel update</p> <ol style="list-style-type: none"> 1. At step 3, user clicks Hủy (Cancel). 2. System discards changes and returns to the profile view page.
Exceptions:	<p>1.0.E1 Validation error</p> <ul style="list-style-type: none"> • If required fields (e.g. full name, email) are missing or invalid, system highlights the errors and prevents saving. <p>1.0.E2 Save failure</p> <ul style="list-style-type: none"> • If the backend fails to persist changes, system displays “Không thể cập nhật hồ sơ. Vui lòng thử lại” and keeps the form open.
Priority:	High
Frequency of Use:	Occasional
Business Rules:	BR-01, BR-22, BR-23
Other Information:	N/A
Assumptions:	User has a stable Internet connection and permissions to update their own profile.

UC-05: Change password

Use Case ID and Name:	UC-05 - Change password		
Created By:	An	Date Created:	09/04/2025
Primary Actor:	Authenticated user	Secondary Actors:	
Trigger:	User clicks “Thay đổi mật khẩu” in their account settings.		
Description:	An authenticated user changes their account password by providing the current password and a new one.		
Preconditions:	PRE-1: User is logged in PRE-2: User knows their current password		
Post-conditions:	POST-1: User’s password is updated and takes effect immediately POST-2: All active sessions (except the current one) are invalidated		
Normal Flow:	1.0 Change password <ol style="list-style-type: none"> 1. User selects “Thay đổi mật khẩu” in the settings menu. 2. System displays the change-password form. 3. User enters the current password, a new password, and confirms the new password. 4. User clicks “Lưu”. 5. System validates the current password and new-password rules, updates the password, invalidates other sessions, and shows “Mật khẩu đã được cập nhật.” 		
Alternative Flows:	1.1 New passwords do not match <ol style="list-style-type: none"> 1. At step 4, if the new-password and confirmation differ, system displays “Mật khẩu mới không khớp” and keeps the form open. 1.2 Weak new password <ol style="list-style-type: none"> 1. At step 4, if the new password fails strength rules, system shows “Mật khẩu phải có ít nhất 8 ký tự, bao gồm chữ hoa, chữ thường và số.” 		
Exceptions:	1.0.E1 Incorrect current password <ol style="list-style-type: none"> 1. At step 5, if the current password is wrong, system displays “Mật khẩu cũ không đúng” and keeps the form open. 1.0.E2 System error		

	<p>1. If the update fails due to server issues, system shows “Không thể thay đổi mật khẩu. Vui lòng thử lại.”</p>
Priority:	High (Medium, Low)
Frequency of Use:	Occasional
Business Rules:	BR-24, BR-25, BR-26
Other Information:	N/A
Assumptions:	<p>User has a stable connection.</p> <p>Password update service is operational.</p>

UC-06: Reset password

Use Case ID and Name:	UC-06 - Reset password		
Created By:	An	Date Created:	09/04/2025
Primary Actor:	Unauthenticated user	Secondary Actors:	
Trigger:	User clicks “Quên mật khẩu?” on the login page.		
Description:	A user who cannot recall their password requests a reset link or OTP via email, then sets a new password to regain access.		
Preconditions:	<p>PRE-1: User has previously registered and verified their email address.</p> <p>PRE-2: Email/OTP service is operational.</p>		
Post-conditions:	<p>POST-1: User's password is updated to the new value.</p> <p>POST-2: User can now authenticate with the new password.</p>		
Normal Flow:	<p>1.0 Reset password</p> <ol style="list-style-type: none"> 1. User clicks “Quên mật khẩu?” on the login screen. 2. System prompts for the registered email address. 3. User enters their email and clicks Gửi mã OTP. 4. System sends a one-time code (OTP) to that email. 5. User enters the received OTP and clicks Xác nhận. 6. System verifies the OTP and displays the new-password form. 7. User enters a new password, confirms it, and clicks Cập nhật mật khẩu. 		

	<p>8. System validates complexity rules, updates the password, and shows “Mật khẩu đã được thay đổi.”</p>
Alternative Flows:	<p>1.1 Email not found</p> <ol style="list-style-type: none"> At step 3, if the email isn't registered, system displays “Email chưa được đăng ký.” and stays on the email prompt. <p>1.2 OTP expired or invalid</p> <ol style="list-style-type: none"> At step 6, if the entered OTP is wrong or has expired, system shows “OTP không hợp lệ hoặc đã hết hiệu lực.” and allows retry.
Exceptions:	<p>1.0.E1 Validation error</p> <ol style="list-style-type: none"> If the new-password and confirmation don't match or fail complexity, system highlights errors and prevents update. <p>1.0.E2 System error</p> <ol style="list-style-type: none"> On server failure (email service or database issue), system displays “Không thể đặt lại mật khẩu. Vui lòng thử lại sau.”
Priority:	High
Frequency of Use:	Occasional
Business Rules:	BR-18, BR-24, BR-27
Other Information:	N/A
Assumptions:	Users have access to their registered email and internet connectivity.

UC-07: View product list

Use Case ID and Name:	UC-07-View product list		
Created By:	An	Date Created:	09/04/2025
Primary Actor:	Guest	Secondary Actors:	
Trigger:	Guest clicks “Sản phẩm” in the main navigation.		
Description:	A guest browses the public catalog of products without logging in.		
Preconditions:	PRE-1: Product listing service is available PRE-2: There is at least one published product		

Post-conditions:	POST-1: The product list is retrieved and shown to the guest
Normal Flow:	<p>1.0 View product list</p> <ol style="list-style-type: none"> 1. Guest selects “Sản phẩm”. 2. System fetches all published products. 3. System displays the list.
Alternative Flows:	<p>1.1 No products available</p> <ol style="list-style-type: none"> 1. System shows “Chưa có sản phẩm để hiển thị.”
Exceptions:	<p>1.0.E1 Retrieval error</p> <ol style="list-style-type: none"> 1. System displays “Không thể tải sản phẩm. Vui lòng thử lại.”
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-28, BR-29, BR-30
Other Information:	N/A
Assumptions:	Guest has a working internet connection.

UC-08: View product detail

Use Case ID and Name:	UC-08 -View product detail		
Created By:	An	Date Created:	09/04/2025
Primary Actor:	Guest	Secondary Actors:	
Trigger:	Guest clicks a product's view action in the public product list.		
Description:	Guest inspects the full details of a publicly available product, including images, description, price, and stock status.		
Preconditions:	PRE-1: Product list has been displayed (UC-07 executed) PRE-2: The selected product is marked “Published”		
Post-conditions:	POST-1: The product's detail page is rendered		
Normal Flow:	<p>1.0 View product detail</p> <ol style="list-style-type: none"> 1. Guest selects “Sản phẩm” in the main navigation. 		

	<ol style="list-style-type: none"> 2. System displays the product list. 3. Guest clicks the view icon on a product. 4. System retrieves and shows that product's full details.
Alternative Flows:	<p>1.1 Product not available</p> <ol style="list-style-type: none"> 1. At step 3, if the product is unpublished or deleted, system displays “Sản phẩm không tồn tại” and returns to the list.
Exceptions:	<p>1.0.E1 Retrieval failure</p> <ol style="list-style-type: none"> 1. If loading the detail fails, system displays “Không thể tải chi tiết sản phẩm. Vui lòng thử lại.”
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-31, BR-32, BR-33
Other Information:	N/A
Assumptions:	Backend services are operational.

UC-09: View woodworker list

Use Case ID and Name:	UC-09-View woodworker list		
Created By:	An	Date Created:	09/04/2024
Primary Actor:	Guest	Secondary Actors:	
Trigger:	Guest clicks “Xưởng mộc” in the main navigation.		
Description:	Guest opens the public directory of all approved woodworkers to browse profiles and specialties.		
Preconditions:	PRE-1: Guest is on the public site. PRE-2: Woodworker directory service is available.		
Post-conditions:	POST-1: The list of approved woodworkers is retrieved and displayed to the guest.		
Normal Flow:	<p>1.0 View woodworker list</p> <ol style="list-style-type: none"> 1. Guest selects “Xưởng mộc” in the main menu. 2. System fetches all approved woodworker profiles. 3. System displays the woodworker list. 		

Alternative Flows:	<p>1.1 No woodworkers available</p> <p>1. System shows “Chưa có xưởng mộc nào” in place of the list.</p>
Exceptions:	<p>1.0.E1 Retrieval error</p> <p>1. If fetching fails, system displays “Không thể tải danh sách xưởng mộc. Vui lòng thử lại.”</p>
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-34, BR-35, BR-36
Other Information:	N/A
Assumptions:	Guest has a working Internet connection.

UC-10: View woodworker detail

Use Case ID and Name:	UC-10 - View woodworker detail		
Created By:	An	Date Created:	09/04/2024
Primary Actor:	Guest	Secondary Actors:	
Trigger:	Guest clicks the view action on a woodworker in the directory.		
Description:	A guest examines the full profile of a woodworker, including their name, rating, specialties, portfolio samples, and contact information.		
Preconditions:	PRE-1: Guest has accessed the woodworker list (UC-09 completed) PRE-2: The selected woodworker exists and has Approved status		
Post-conditions:	POST-1: The woodworker's detail page is rendered POST-2: No changes are made to any records		
Normal Flow:	<p>1.0 View woodworker detail</p> <ol style="list-style-type: none"> 1. Guest selects “Xưởng mộc” in the main navigation. 2. System displays the list of approved woodworkers. 3. Guest clicks the view icon on the chosen woodworker entry. 4. System retrieves that woodworker's full profile and displays the detail page. 		

Alternative Flows:	<p>1.1 Woodworker not found</p> <p>1. At step 3, if the woodworker has been removed or is unapproved, system shows “Không tìm thấy Xưởng mộc” and returns to the directory.</p>
Exceptions:	<p>1.0.E1 Data retrieval error</p> <p>1. If fetching the profile fails, system displays “Không thể tải thông tin Xưởng mộc. Vui lòng thử lại.”</p>
Priority:	High (Medium, Low)
Frequency of Use:	Frequent
Business Rules:	BR-34, BR-37, BR-38
Other Information:	N/A
Assumptions:	Backend profile service is operational and network connectivity is stable.

UC-11: View design idea list

Use Case ID and Name:	UC-11-View design idea list		
Created By:	An	Date Created:	09/04/2025
Primary Actor:	Guest	Secondary Actors:	
Trigger:	Guest clicks “Ý tưởng thiết kế” in the main navigation.		
Description:	Guest browses the catalog of all publicly available design ideas.		
Preconditions:	PRE-1: Design idea service is available PRE-2: There is at least one published design idea		
Post-conditions:	POST-1: The list of design ideas is retrieved and displayed		
Normal Flow:	<p>1.0 View design idea list</p> <ol style="list-style-type: none"> Guest selects “Ý tưởng thiết kế”. System retrieves all published design ideas. System displays the design idea list. 		
Alternative Flows:	1.1 No design ideas available		

	1. System shows “Chưa có ý tưởng thiết kế nào.”
Exceptions:	<p>1.0.E1 Retrieval error</p> <p>1. System displays “Không thể tải danh sách ý tưởng thiết kế. Vui lòng thử lại.”</p>
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-39, BR-40, BR-41
Other Information:	N/A
Assumptions:	Guest has a stable Internet connection.

UC-12: View design idea detail

Use Case ID and Name:	UC-12-View design idea detail		
Created By:	An	Date Created:	09/04/2025
Primary Actor:	Guest	Secondary Actors:	
Trigger:	Guest clicks the view icon on a design idea in the list.		
Description:	Guest inspects the full content of a publicly available design idea, including images, description, materials, and creator notes.		
Preconditions:	<p>PRE-1: The design idea's detail page is rendered</p> <p>PRE-2: No modification to any data</p>		
Post-conditions:	<p>POST-1: The design idea's detail page is rendered</p> <p>POST-2: No modification to any data</p>		
Normal Flow:	<p>1.0 View design idea detail</p> <ol style="list-style-type: none"> 1. Guest selects “Ý tưởng thiết kế” in the main menu. 2. System displays the list of published design ideas. 3. Guest clicks the view icon on a specific design idea. 4. System retrieves and shows that idea's full details. 		
Alternative Flows:	1.1 Idea not available		

	<p>1. At step 3, if the idea is unpublished or deleted, system displays “Ý tưởng không tồn tại” and returns to the list.</p>
Exceptions:	<p>1.0.E1 Retrieval failure</p> <p>1. If loading the detail fails, system displays “Không thể tải chi tiết ý tưởng. Vui lòng thử lại.”</p>
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-39, BR-42, BR-43
Other Information:	N/A
Assumptions:	Backend services and network connectivity are operational.

UC-13: View service order history

Use Case ID and Name:	UC-13 -View service order history		
Created By:	An	Date Created:	09/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer clicks “Lịch sử đơn hàng” in their account menu.		
Description:	Customer reviews the full history of all service orders they've ever placed (e.g., repairs, guarantees, installations).		
Preconditions:	<p>PRE-1: Customer is logged in</p> <p>PRE-2: At least one service order exists for that customer</p>		
Post-conditions:	POST-1: The customer's service order list is retrieved and displayed		
Normal Flow:	<p>1.0 Display service order history</p> <ol style="list-style-type: none"> 1. Customer selects “Lịch sử đơn dịch vụ”. 2. System retrieves all service orders tied to that customer. 3. System displays the list, ordered by most recent first. 		
Alternative Flows:	1.1 No service orders found		

	1. System shows “Bạn chưa có đơn dịch vụ nào” in place of the list.
Exceptions:	<p>1.0.E1 Retrieval error</p> <p>1. If fetching fails, system displays “Không thể tải lịch sử đơn dịch vụ. Vui lòng thử lại.”</p>
Priority:	Medium
Frequency of Use:	Occasional
Business Rules:	BR-44, BR-45, BR-46
Other Information:	N/A
Assumptions:	Assumptions: Backend service-order endpoint is operational.

UC-14: View service order detail

Use Case ID and Name:	UC-14-View service order detail		
Created By:	An	Date Created:	09/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer clicks “Xem chi tiết” on a service order in their history.		
Description:	Customer inspects the full details of a selected service order, including its type, creation date, status timeline, cost, attachments, and any notes or responses.		
Preconditions:	<p>PRE-1: Customer is logged in</p> <p>PRE-2: The selected service order exists and belongs to that customer</p>		
Post-conditions:	<p>POST-1: The service order's details are presented on-screen</p> <p>POST-2: No change is made to the order's data</p>		
Normal Flow:	<p>1.0 Display service order detail</p> <ol style="list-style-type: none"> 1. Customer selects “Xem chi tiết” on a service order entry. 2. System retrieves the corresponding service order record. 3. System displays all details: order ID, type, dates, status history, cost, attachments, and staff notes. 		
Alternative Flows:	1.1 Order not found or unauthorized		

	<p>1. At step 2, if the record is missing or not owned by the customer, system shows “Không tìm thấy đơn dịch vụ” and returns to the history list.</p>
Exceptions:	<p>1.0.E1 Data retrieval error</p> <p>1. If the system fails to load the order details, display “Không thể tải chi tiết đơn dịch vụ. Vui lòng thử lại.”</p>
Priority:	Medium
Frequency of Use:	Occasional
Business Rules:	BR-47, BR-48, BR-49
Other Information:	N/A
Assumptions:	Service-order API is operational and network connectivity is stable.

UC-15: View guarantee order history

Use Case ID and Name:	UC-15 – View guarantee order history		
Created By:	An	Date Created:	09/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer clicks “Lịch sử bảo hành” in their account menu.		
Description:	Customer reviews the full list of guarantee/repair requests they have submitted.		
Preconditions:	<p>PRE-1: Customer is logged in</p> <p>PRE-2: Customer has ≥ 1 guarantee/repair request in their history</p>		
Post-conditions:	POST-1: The customer’s guarantee order list is retrieved and shown		
Normal Flow:	<p>1.0 Display guarantee history</p> <ol style="list-style-type: none"> Customer selects “Lịch sử bảo hành”. System retrieves all guarantee/repair orders for that customer. System displays the retrieved list. 		
Alternative Flows:	<p>1.1 No guarantee orders found</p> <ol style="list-style-type: none"> System shows “Bạn chưa có đơn bảo hành nào.” 		

Exceptions:	1.0.E1 Retrieval error 1. System displays “Không thể tải danh sách bảo hành. Vui lòng thử lại.”
Priority:	Medium
Frequency of Use:	Occasional
Business Rules:	BR-52, BR-53, BR-54
Other Information:	N/A
Assumptions:	Guarantee service endpoint is available and responsive.

UC-16: View guarantee order detail

Use Case ID and Name:	UC-16 - View guarantee order detail		
Created By:	An	Date Created:	10/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer clicks “Xem chi tiết” on a guarantee/repair order in their history.		
Description:	Customer inspects the full details of a selected guarantee/repair request, including its type, submission date, current status timeline, any uploaded attachments (photos/docs), cost estimates, and staff notes or responses.		
Preconditions:	PRE-1: Customer is logged in PRE-2: The customer’s guarantee order history page has been displayed (UC-15 executed) PRE-3: The selected guarantee order exists and belongs to the customer		
Post-conditions:	POST-1: The guarantee order’s full details are presented on screen POST-2: No change is made to the order record		
Normal Flow:	1.0 Display guarantee order detail 1. Customer selects “Lịch sử bảo hành” from their account menu. 2. System displays the list of guarantee/repair orders. 3. Customer clicks “Xem chi tiết” on a specific order. 4. System retrieves that order’s full record (type, request date, status timeline, cost, attachments, staff notes) and displays the detail page.		

Alternative Flows:	<p>1.1 Order not found or unauthorized</p> <ol style="list-style-type: none"> At step 3, if the order ID is invalid or does not belong to the customer, system shows “Không tìm thấy đơn bảo hành” and returns to the history list.
Exceptions:	<p>1.0.E1 Data retrieval error</p> <ol style="list-style-type: none"> If the system fails to load the order details, display “Không thể tải chi tiết bảo hành. Vui lòng thử lại.”
Priority:	Medium
Frequency of Use:	Occasional
Business Rules:	BR-55, BR-56, BR-57
Other Information:	N/A
Assumptions:	System services for guarantee order retrieval are operational.

UC-17: View address list

Use Case ID and Name:	UC-17- View address list		
Created By:	An	Date Created:	10/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer selects “Quản lý địa chỉ” from their account/profile menu.		
Description:	Customer reviews all saved shipping addresses in their profile.		
Preconditions:	PRE-1: Customer is logged in		
Post-conditions:	POST-1: The customer’s address list is retrieved and displayed		
Normal Flow:	<p>1.0 View address list</p> <ol style="list-style-type: none"> Customer clicks “Quản lý địa chỉ”. System fetches the customer’s saved addresses. System presents the list of addresses. 		
Alternative Flows:	<p>1.1 No addresses found</p> <ol style="list-style-type: none"> If the customer has no saved addresses, system shows “Chưa có địa chỉ nào” and offers a “Thêm địa chỉ mới” button. 		

Exceptions:	<p>1.0.E1 Retrieval error</p> <p>1. If fetching fails, system displays “Không thể tải danh sách địa chỉ. Vui lòng thử lại.”</p>
Priority:	Medium
Frequency of Use:	Occasional
Business Rules:	BR-58, BR-59, BR-60
Other Information:	N/A
Assumptions:	Network connectivity and address-service endpoint are available.

UC-18: Add new address

Use Case ID and Name:	UC-18-Add new address		
Created By:	An	Date Created:	10/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer clicks “Thêm địa chỉ mới” on their address list page.		
Description:	Customer adds a new shipping address to their profile for future orders.		
Preconditions:	PRE-1: Customer is logged in PRE-2: Customer has navigated to the address management screen		
Post-conditions:	POST-1: The new address is saved and appears in the customer’s address list		
Normal Flow:	<p>1.0 Add new address</p> <ol style="list-style-type: none"> Customer clicks “Thêm địa chỉ mới”. System displays the new-address form. Customer fills in recipient name, phone, and full address, then clicks Lưu. System validates the input, saves the address, and shows a confirmation message. 		
Alternative Flows:	<p>1.1 Missing required fields</p> <ol style="list-style-type: none"> At step 3, if any mandatory field is empty, system highlights the missing entries and prevents saving. 		

Exceptions:	<p>1.0.E1 Save failure</p> <p>1. If the system cannot save the address (e.g. server error), it displays “Không thể thêm địa chỉ. Vui lòng thử lại.”</p>
Priority:	High
Frequency of Use:	Occasional
Business Rules:	BR-61, BR-62, BR-63
Other Information:	N/A
Assumptions:	Address-service endpoint is available and network connection is stable.

UC-19: Update address

Use Case ID and Name:	UC-19 - Update address		
Created By:	An	Date Created:	10/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer clicks “Chỉnh sửa” beside a saved address on their address list screen.		
Description:	Customer modifies the details of an existing shipping address in their profile.		
Preconditions:	<p>PRE-1: Customer is logged in</p> <p>PRE-2: The address list page is displayed, and at least one address exists</p>		
Post-conditions:	POST-1: The selected address is updated and reflected in the list		
Normal Flow:	<p>1.0 Update address</p> <ol style="list-style-type: none"> 1. Customer selects “Quản lý địa chỉ”. 2. System displays the customer’s saved addresses. 3. Customer clicks “Chỉnh sửa” on the desired address. 4. System opens the update form populated with the current address details. 5. Customer edits fields (recipient, phone, location details) and clicks “Lưu”. 6. System validates the input, saves the changes, and shows “Cập nhật địa chỉ thành công.” 		
Alternative Flows:	<p>1.1 Cancel update</p> <ol style="list-style-type: none"> 1. At step 5, customer clicks “Hủy”. 		

	<p>2. System discards any edits and returns to the address list.</p>
Exceptions:	<p>1.0.E1 Validation error</p> <p>1. If mandatory fields are empty or phone format is invalid, system highlights errors and prevents saving.</p> <p>1.0.E2 Save failure</p> <p>1. If the system cannot persist the update, it displays “Không thể cập nhật địa chỉ. Vui lòng thử lại.”</p>
Priority:	High
Frequency of Use:	Occasional
Business Rules:	BR-61, BR-63, BR-64
Other Information:	N/A
Assumptions:	<p>1. Address-management service is available and responsive.</p> <p>2. Customer has a stable Internet connection.</p>

UC-20: Buy product

Use Case ID and Name:	UC-20-Buy product		
Created By:	An	Date Created:	11/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer decides to purchase a product.		
Description:	Customer browses products, adds a chosen item to their cart, completes checkout, and receives an order confirmation.		
Preconditions:	PRE-1. Customer is logged in PRE-2. Customer has at least one saved shipping address PRE-3. The product has sufficient stock		
Post-conditions:	POST-1: A new order record is created POST-2: Product stock levels are decremented POST-3: Customer sees an order confirmation page		
Normal Flow:	1.0 Complete a purchase <ol style="list-style-type: none"> 1. Customer clicks “Sản phẩm” in the main menu. 2. System displays the product list. 3. Customer selects a product card. 4. System shows the product detail page. 		

	<p>5. Customer clicks “Thêm vào giỏ”.</p> <p>6. System confirms addition and opens the cart sidebar.</p> <p>7. Customer reviews their cart, selects a shipping address, enters any order notes, and clicks “Tiến hành đặt hàng”.</p> <p>8. System validates the cart and address, creates the order, and redirects to a success screen.</p> <p>9. System displays “Đặt hàng thành công” with options to return home or view order history.</p>
Alternative Flows:	<p>1.1 Immediate purchase</p> <ol style="list-style-type: none"> At step 5, customer clicks “Mua ngay” instead. System skips the cart sidebar and shows the checkout form directly. Continue at step 7. <p>1.2 No saved address</p> <ol style="list-style-type: none"> At step 7, if no address exists, system prompts customer to add one before proceeding.
Exceptions:	<p>1.0.E1 Insufficient stock</p> <ol style="list-style-type: none"> At step 6, if stock is insufficient, system shows “Số lượng không đủ” and blocks checkout. <p>1.0.E2 Validation error</p> <ol style="list-style-type: none"> At step 7, missing or invalid inputs trigger inline errors and prevent submission. <p>1.0.E3 System failure</p> <ol style="list-style-type: none"> If order creation fails, system displays “Không thể hoàn tất đặt hàng. Vui lòng thử lại.”
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-08, BR-65, BR-66, BR-71
Other Information:	N/A
Assumptions:	Payment is handled on delivery.

UC-21: Book a personalization service

Use Case ID and Name:	UC-21 – Book a personalization service		
Created By:	An	Date Created:	12/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer decides to request a personalized product from a woodworker.		
Description:	Customer browses woodworkers, selects the personalization service, configures one or more custom products (dimensions, material, finish, special requests, sample images), chooses shipping options, and submits the personalization order.		

Preconditions:	PRE-1: Customer is logged in PRE-2: Customer has at least one saved shipping address PRE-3: Personalization service is available for the selected woodworker
Post-conditions:	POST-1: A new personalization service request is created POST-2: Each custom product is stored with its specifications POST-3: The woodworker is notified of the new request
Normal Flow:	<p>1.0 Book personalization service</p> <ol style="list-style-type: none"> 1. Customer clicks “Xưởng mộc” in the main menu. 2. System displays the list of woodworkers. 3. Customer selects a woodworker’s card. 4. System opens the woodworker’s detail page. 5. Customer clicks the “Dịch vụ cá nhân hóa” tab. 6. System shows the personalization form with fields for category, dimensions (D×R×C), wood type, surface finish, special requests, sample image upload (up to 4), and quantity. 7. Customer fills the form and clicks “+ Thêm sản phẩm”. 8. System adds the custom product to the right-hand list (max 4 items). 9. Customer repeats steps 6–8 for additional items or proceeds. 10. Customer selects a shipping address, enters any order notes, checks “Yêu cầu giao hàng + lắp đặt bởi xưởng”, then clicks “Gửi yêu cầu”. 11. System validates entries, creates the service order, and redirects to the confirmation page. 12. System displays “Đặt hàng thành công” with options to return home or view order history.
Alternative Flows:	<p>1.1 No personalization items added</p> <ol style="list-style-type: none"> 1. At step 10, if the customer hasn’t added any product, system shows “Vui lòng thêm ít nhất một sản phẩm” and blocks submission.
Exceptions:	<p>1.0.E1 Missing required fields</p> <ol style="list-style-type: none"> 1. At step 6, incomplete fields trigger inline errors and prevent “+ Thêm sản phẩm” or “Gửi yêu cầu”. <p>1.0.E2 Image upload failure</p> <ol style="list-style-type: none"> 1. If a sample image exceeds 5 MB or is an unsupported format, system rejects it with “Không thể tải ảnh.” <p>1.0.E3 Submission error</p> <ol style="list-style-type: none"> 1. If order creation fails, system displays “Không thể gửi yêu cầu. Vui lòng thử lại.”
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-65, BR-67, BR-68, BR-08, BR-71
Other Information:	N/A
Assumptions:	Backend service for personalization orders is available and responsive.

UC-22: Book a customization service

Use Case ID and Name:	UC-22 – Book a customization service		
Created By:	An	Date Created:	12/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer decides to order a designer's ready-made design with custom configuration.		
Description:	Customer browses available designs, configures a chosen design (size, wood type), adds it to their cart (or buys immediately), completes checkout with a shipping address and notes, and receives confirmation.		
Preconditions:	PRE-1: Customer is logged in PRE-2: Customer has at least one saved shipping address PRE-3: Design customization service is enabled for the selected design		
Post-conditions:	POST-1: A new customization order is created POST-2: Each configured design item is stored with its specifications POST-3: The woodworker is notified of the new order request		
Normal Flow:	1.0 Book a customization service <ol style="list-style-type: none"> 1. Customer clicks “Thiết kế” in the main menu. 2. System displays the design gallery. 3. Customer selects a design card. 4. System opens the design detail page. 5. Customer configures the design (selects dimensions and wood type). 6. Customer clicks “Thêm vào giỏ”. 7. System confirms addition and opens the cart sidebar under the Thiết kế tab. 8. Customer reviews their cart, selects a shipping address, enters any order notes, and clicks “Tiến hành đặt hàng”. 9. System validates inputs, creates the customization order, and redirects to a success page. 10. System displays “Đặt hàng thành công” with options to return home or view order history. 		
Alternative Flows:	1.1 Immediate order <ol style="list-style-type: none"> 1. At step 6, instead of “Thêm vào giỏ”, customer clicks “ĐẶT NGAY”. 2. System skips the cart sidebar and goes straight to the checkout form. 3. Continue at step 8. 1.2 No saved address <ol style="list-style-type: none"> 1. At step 8, if the customer has no address, system prompts to add one before proceeding. 		
Exceptions:	1.0.E1 Missing configuration		

	<p>1. At step 6, if required configuration fields are empty, system highlights them and prevents addition.</p> <p>1.0.E2 Image upload failure</p> <ol style="list-style-type: none"> If a user-provided sample image exceeds 5 MB or is unsupported, system rejects it with “Không thể tải ảnh.” <p>1.0.E3 Submission error</p> <ol style="list-style-type: none"> If order creation fails, system displays “Không thể hoàn tất đặt hàng. Vui lòng thử lại.”
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-65, BR-67, BR-08, BR-71
Other Information:	N/A
Assumptions:	Backend service for customization orders is available and responsive.

UC-23: Book a guarantee service

Use Case ID and Name:	UC-23 - Book a guarantee service		
Created By:	Luân	Date Created:	24/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	The customer wants to request a guarantee or repair service for a purchased product.		
Description:	This use case describes how a logged-in customer selects one of their completed orders, chooses a specific product, specifies the type of issue and its details, uploads supporting images, selects a shipping address, and submits a guarantee (warranty/repair) request.		
Preconditions:	PRE-1. Customer is logged in. PRE-2. Customer has at least one completed order in their order history. PRE-3. Customer has at least one saved shipping address. PRE-4. The selected product is eligible for guarantee or repair service.		
Post-conditions:	POST-1. A guarantee service request is created with status Pending. POST-2. The customer receives a confirmation notification.		
Normal Flow:	1.0 Book guarantee/repair service <ol style="list-style-type: none"> Customer clicks “Sửa chữa / bảo hành” in the nav bar System displays the request page with: <ol style="list-style-type: none"> “Chọn đơn hàng đã hoàn thành” dropdown “Địa chỉ giao hàng” panel Disabled “Gửi yêu cầu sửa chữa” button Customer opens the order dropdown and picks one completed order System loads that order (number, date, total) and enables “Chọn sản phẩm...” 		

	<p>5. Customer clicks “Chọn sản phẩm cần sửa chữa / bảo hành”</p> <p>6. System shows each line-item (image, date, warranty period, expiry)</p> <p>7. Customer selects the item and system highlights it + shows the woodworker card</p> <p>8. Customer selects a shipping address (or clicks “Quản lý địa chỉ” to add one)</p> <p>9. System renders the condition form with:</p> <ul style="list-style-type: none"> 9.1 “BẢO HÀNH” or “SỬA CHỮA” badge 9.2 “Loại lỗi bảo hành” dropdown 9.3 “Mô tả tình trạng hiện tại” textarea 9.4 “Hình ảnh tình trạng hiện tại” upload box <p>10. Customer chooses defect type, writes details, and uploads up to 3 pics</p> <p>11. System validates inputs and enables “Gửi yêu cầu sửa chữa”</p> <p>12. Customer hits “Gửi yêu cầu sửa chữa” → system re-validates, creates the request, sends notifications</p>
Alternative Flows:	<p>1.1 No completed orders</p> <ol style="list-style-type: none"> 1. Dropdown reads “Không có đơn hàng hoàn thành” 2. System suggests: “Xem lịch sử đơn hàng” <p>1.2 No saved addresses</p> <ol style="list-style-type: none"> 1. Address panel reads “Chưa có địa chỉ giao hàng” + link to manage 2. Customer adds address, returns to form <p>1.3 Warranty expired</p> <ol style="list-style-type: none"> 1. Badge auto-switches to “SỬA CHỮA” 2. Customer fills repair details as in 1.0
Exceptions:	<p>1.0.E1 Validation error</p> <ol style="list-style-type: none"> 1. Missing/invalid inputs trigger inline errors; submission stays disabled <p>1.0.E2 Image upload failure</p> <ol style="list-style-type: none"> 1. Unsupported format or file >5 MB shows an upload error, rejects file <p>1.0.E3 System error</p> <ol style="list-style-type: none"> 1. Server failure shows generic error prompt and “Please try again”
Priority:	High
Frequency of Use:	Occasional
Business Rules:	BR-04, BR-05, BR-06, BR-07, BR-08, BR-71
Other Information:	N/A
Assumptions:	Customers have stable Internet for uploads

UC-24: Add feedback to order detail

Use Case ID and Name:	UC-24: Add feedback to order detail		
Created By:	Luân	Date Created:	10/04/2025

Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer clicks “Gửi phản hồi” on the order detail page		
Description:	On an order that's awaiting customer approval (e.g. contract, design, or schedule), the customer can send targeted feedback back to the woodworker requesting changes or clarifications.		
Preconditions:	PRE-1. Customer is logged in PRE-2. The order is currently awaiting customer approval (e.g. contract / design/ appointment) PRE-3. No feedback has been sent for this approval step yet		
Post-conditions:	POST-1. The customer's feedback is saved and marked as “awaiting woodworker action” POST-2. A notification is dispatched to the woodworker POST-3. The order remains in the approval flow until the woodworker addresses the feedback		
Normal Flow:	1.0 Send feedback <ol style="list-style-type: none"> 1. Customer clicks Gửi phản hồi in the order header. 2. System opens a “Gửi phản hồi” modal containing: an input for feedback and information need to be confirmed 3. Customer selects a feedback type, writes their comments, and (optionally) attaches a file. 4. Once the text area is non-empty, Gửi phản hồi becomes enabled. 5. Customer clicks Gửi phản hồi. 6. System validates inputs, saves the feedback record in “pending woodworker action,” closes the modal, and displays “Phản hồi đã được gửi.” 7. Woodworker receives a notification to review the customer's feedback and make the requested changes. 		
Alternative Flows:	N/A		
Exceptions:	N/A		
Priority:	High		
Frequency of Use:	Occasional		
Business Rules:	BR-12, BR-71		
Other Information:	N/A		
Assumptions:	N/A		

UC-25: Add review and rating

Use Case ID and Name:	UC-25 – Add review and rating		
Created By:	Luân	Date Created:	25/04/2025

Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer clicks the “Đánh giá” button on the detail page of a completed order.		
Description:	A logged-in customer submits a star rating and written feedback on a completed order. The review is stored in a “Pending” state and must be approved by platform staff before it becomes visible.		
Preconditions:	PRE-1. Customer is logged in PRE-2. Order detail page of an order with status “Đã hoàn tất” is displayed PRE-3. No existing feedback for this order		
Post-conditions:	POST-1. A review record is created POST-1. Customer sees a submission confirmation		
Normal Flow:	<p>1.0 Add feedback</p> <ol style="list-style-type: none"> 1. Customer clicks Đánh giá on the order detail header. 2. System pops up the “Đánh giá đơn hàng” modal containing: 3. A 1–5 star rating control 4. A “Nội dung đánh giá” textarea 5. Buttons Gửi đánh giá (disabled) and Đóng 6. Customer selects a star rating and enters their comments. 7. System enables Gửi đánh giá once both a rating and non-empty comment are provided. 8. Customer clicks Gửi đánh giá. 9. System validates inputs, creates the review record with status = PendingApproval, closes the modal, and shows a “Cảm ơn đánh giá của bạn!” message. 		
Alternative Flows:	N/A		
Exceptions:	N/A		
Priority:	Medium		
Frequency of Use:	Occasional		
Business Rules:	BR-09, BR-10, BR-11, BR-71		
Other Information:	N/A		
Assumptions:	Customers have a stable Internet connection		

UC-26: View complaint list

Use Case ID and Name:	UC-26 – View complaint list		
Created By:	Luân	Date Created:	10/04/2025
Primary Actor:	Customer	Secondary Actors:	

Trigger:	Customer clicks Khiếu nại in the sidebar menu.
Description:	A customer can open their complaint management screen to see all complaints they've submitted, filter by status, and sort by complaint ID.
Preconditions:	PRE-1. Customer is logged in PRE-2. Complaint service is available
Post-conditions:	POST-1. The customer's complaint list is retrieved and displayed POST-2. Any chosen filter and sort preferences are applied
Normal Flow:	<p>1.0 Display complaint list</p> <ol style="list-style-type: none"> 1. Customer navigates to the complaint management screen. 2. System retrieves all complaints submitted by that customer. 3. System applies default or previously selected filter and sort settings. 4. System displays the resulting list of complaints.
Alternative Flows:	N/A
Exceptions:	N/A
Priority:	Medium
Frequency of Use:	Occasional
Business Rules:	BR-71
Other Information:	N/A
Assumptions:	User's network connection is stable.

UC-27: Create complaint

Use Case ID and Name:	UC-27 – Create complaint		
Created By:	Luân	Date Created:	10/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer clicks “Tạo khiếu nại mới” on the complaint list screen.		
Description:	A customer files a new complaint about a order.		
Preconditions:	PRE-1. Customer is logged in PRE-2. Customer has at least one order		
Post-conditions:	POST-1. A new complaint is recorded in “pending review” status		
Normal Flow:	<p>1.0 Create complaint</p> <ol style="list-style-type: none"> 1. Customer clicks “Tạo khiếu nại mới”. 2. Customer fills in the complaint form and submits. 3. System validates the input, saves the complaint, and shows a confirmation message. 		
Alternative Flows:	N/A		

Exceptions:	N/A		
Priority:	Medium		
Frequency of Use:	Occasional		
Business Rules:	BR-71		
Other Information:	N/A		
Assumptions:	User's network connection is stable.		

UC-28: View complaint detail

Use Case ID and Name:	UC-28 – View complaint detail		
Created By:	Luân	Date Created:	10/04/2025
Primary Actor:	Customer	Secondary Actors:	
Trigger:	Customer clicks “Xem chi tiết” on a complaint in their list.		
Description:	Customer examines the full details of one of their submitted complaints, including type, description, attachments, status history, and any responses from the woodworker or support staff.		
Preconditions:	PRE-1. Customer is logged in PRE-2. The selected complaint exists and belongs to that customer		
Post-conditions:	POST-1. Complaint details are presented on-screen		
Normal Flow:	1.0 View complaint detail Customer selects “Xem chi tiết” on a complaint entry. System loads the complaint record and displays all fields and history.		
Alternative Flows:	N/A		
Exceptions:	N/A		
Priority:	Medium		
Frequency of Use:	Occasional		
Business Rules:	BR-13, BR-71		
Other Information:	N/A		
Assumptions:	User's network connection is stable.		

UC-29: View product list

Use Case ID and Name:	UC-29 – View product list		
Created By:	Luân	Date Created:	10/04/2025

Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker clicks “Sản phẩm” in the workshop menu.		
Description:	The woodworker opens their product management screen to see all their products in a sortable, filterable table.		
Preconditions:	PRE-1. Woodworker is logged in and has an active workshop profile PRE-2. Product service is available		
Post-conditions:	POST-1. The woodworker’s full product list is retrieved and rendered POST-2. Any default or previously selected filters/sorts are applied		
Normal Flow:	1.0 Display product list 1. Woodworker selects “Sản phẩm” in the side menu. 2. System fetches all products belonging to that woodworker. 3. System applies default sort (Mã SP descending) and any saved filters. 4. System renders the table of products.		
Alternative Flows:	N/A		
Exceptions:	N/A		
Priority:	High		
Frequency of Use:	Frequent		
Business Rules:	BR-14, BR-70		
Other Information:	N/A		
Assumptions:	User's network connection is stable.		

UC-30: View product detail

Use Case ID and Name:	UC-30 – View product detail		
Created By:	Luân	Date Created:	11/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker clicks the “view” action on a product in their list.		
Description:	Woodworker examines all information for a single product they own.		
Preconditions:	PRE-1. Woodworker is logged in		
Post-conditions:	POST-1. The selected product’s details are loaded and presented		
Normal Flow:	1.0 View product detail 1. Woodworker clicks “Sản phẩm” in the workshop menu. 2. System displays the product list. 3. Woodworker clicks the view icon on the desired product.		

	4. System retrieves that product's full information and displays the detail page.
Alternative Flows:	N/A
Exceptions:	N/A
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-14, BR-70
Other Information:	N/A
Assumptions:	User's network connection is stable.

UC-31: Add product

Use Case ID and Name:	UC-31 – Add product		
Created By:	Luân	Date Created:	10/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker clicks “Thêm sản phẩm mới” on the product list screen.		
Description:	Woodworker adds a new product to their catalog by completing and submitting the product creation form.		
Preconditions:	PRE-1. Woodworker is logged in		
Post-conditions:	POST-1. The new product is saved and appears in the product list		
Normal Flow:	1.0 Add a new product <ol style="list-style-type: none"> 1. Woodworker clicks “Sản phẩm” in the workshop menu. 2. System displays the product list. 3. Woodworker clicks “Thêm sản phẩm mới”. 4. System displays the product creation form. 5. Woodworker fills in the form and submits. 6. System validates the data, saves the product, and confirms creation. 		
Alternative Flows:	1.1 Missing required fields <ol style="list-style-type: none"> 1. At step 3, if required entries are blank, system highlights them and prevents submission. 		
Exceptions:	1.0.E1 Save failure <ol style="list-style-type: none"> 1. If saving fails, system shows “Không thể thêm sản phẩm. Vui lòng thử lại.” 		
Priority:	High		
Frequency of Use:	Occasional		
Business Rules:	BR-15, BR-70		

Other Information:	N/A
Assumptions:	User's network connection is stable.

UC-32: Update product

Use Case ID and Name:	UC-32 – Update product		
Created By:	Luân	Date Created:	10/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker clicks the edit icon on a product in their list.		
Description:	Woodworker modifies the details of an existing product.		
Preconditions:	PRE-1: Woodworker is logged in PRE-2: The product exists and belongs to that woodworker		
Post-conditions:	POST-1: Product record is updated with the new information		
Normal Flow:	1.0 Update product <ol style="list-style-type: none"> 1. Woodworker clicks “Sản phẩm” in the workshop menu. 2. System displays the product list. 3. Woodworker clicks the edit icon next to a product. 4. System displays the product update form populated with current values. 5. Woodworker changes fields as needed and submits. 6. System validates inputs, saves changes, and confirms the update. 		
Alternative Flows:	1.1 No changes made <ol style="list-style-type: none"> 1. At step 3, if the woodworker makes no edits and submits, system detects no differences, skips save, and shows “Không có thay đổi.” 		
Exceptions:	1.0.E1 Validation error <ol style="list-style-type: none"> 1. If any required field is invalid or empty, system highlights errors and prevents submission. 1.0.E2 Save failure <ol style="list-style-type: none"> 1. If saving fails, system displays “Không thể cập nhật sản phẩm. Vui lòng thử lại.” 		
Priority:	High		
Frequency of Use:	Occasional		
Business Rules:	BR-15, BR-70		
Other Information:	N/A		
Assumptions:	Update service is available and responsive.		

UC-33: Delete product

Use Case ID and Name:	UC-33 – Delete product		
Created By:	Luân	Date Created:	11/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker chooses to remove a product from their catalog.		
Description:	Woodworker permanently deletes an existing product from their workshop's product list.		
Preconditions:	PRE-1: Woodworker is logged in PRE-2: The product exists and belongs to that woodworker		
Post-conditions:	POST-1: The product is removed and no longer appears in the list		
Normal Flow:	1.0 Delete a product 1. Woodworker clicks “Sản phẩm” in the workshop menu. 2. System displays the product list. 3. Woodworker clicks the delete icon on the desired product. 4. System prompts “Bạn có chắc chắn muốn xóa sản phẩm này?” 5. Woodworker confirms deletion. 6. System removes the product and displays “Xóa thành công.”		
Alternative Flows:	1.1 Cancellation 1. At step 4, woodworker selects Hủy. 2. System aborts the deletion and returns to the list.		
Exceptions:	1.0.E1 Deletion error 1. If removal fails, system shows “Không thể xóa sản phẩm. Vui lòng thử lại.”		
Priority:	Low		
Frequency of Use:	Occasional		
Business Rules:	BR-16, BR-69, BR-70		
Other Information:	N/A		
Assumptions:	User's network connection is stable.		

UC-34: View post list

Use Case ID and Name:	UC-34 – View post list		
Created By:	Luân	Date Created:	13/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker clicks “Bài đăng” in the workshop menu.		

Description:	Woodworker opens the post management screen to see all blog/articles they have created, with the ability to filter, sort, and navigate to individual posts.
Preconditions:	PRE-1: Woodworker is logged in PRE-2: The post management service is available
Post-conditions:	POST-1: The system retrieves and displays the woodworker's list of posts POST-2: Default filter and sort settings are applied
Normal Flow:	<p>1.0 Display post list</p> <ol style="list-style-type: none"> 1. Woodworker selects “Bài đăng”. 2. System fetches all posts authored by that woodworker. 3. System displays the list of posts.
Alternative Flows:	<p>1.1 No posts found</p> <ol style="list-style-type: none"> 1. If the woodworker has not created any posts, system shows “Chưa có bài viết nào.”
Exceptions:	<p>1.0.E1 Retrieval error</p> <ol style="list-style-type: none"> 1. If the system fails to load posts, display “Không thể tải danh sách bài viết. Vui lòng thử lại.”
Priority:	Medium
Frequency of Use:	Occasional
Business Rules:	BR-70
Other Information:	N/A
Assumptions:	Network connection and backend service are functional.

UC-35: View post detail

Use Case ID and Name:	UC-35 – View post detail		
Created By:	Luân	Date Created:	14/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker clicks on a specific post in the post list to view its detailed content.		
Description:	The woodworker views the detailed content of a specific post, which includes the full text, images, and related information.		
Preconditions:	PRE-1: Woodworker is logged in. PRE-2: The post the woodworker wishes to view exists.		
Post-conditions:	POST-1: The system retrieves and displays the detailed content of the selected post. POST-2: Woodworker can see options to edit or delete the post if they are the author.		

Normal Flow:	<p>1.0 Display post details</p> <ol style="list-style-type: none"> 1. Woodworker selects “Bài đăng”. 2. System fetches all posts authored by that woodworker. 3. Woodworker clicks on a post from the list. 4. System fetches and displays the detailed content of the selected post.
Alternative Flows:	<p>1.1 No post found</p> <ol style="list-style-type: none"> 1. If the post cannot be found, system displays “Bài viết không tồn tại.”
Exceptions:	<p>1.0.E1 Retrieval error</p> <ol style="list-style-type: none"> 1. If the system fails to load the post details, display “Không thể tải bài viết. Vui lòng thử lại.”
Priority:	Medium
Frequency of Use:	Occasional
Business Rules:	BR-70
Other Information:	N/A
Assumptions:	Network connection and backend service are functional.

UC-36: Create post

Use Case ID and Name:	UC-36 – Create post		
Created By:	Luân	Date Created:	12/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker clicks on “Thêm bài viết mới” after viewing the post list to create a new post.		
Description:	Woodworker creates a new post on the platform, entering the required title, content, and optional images. The post is then added to the list of posts.		
Preconditions:	<p>PRE-1: Woodworker is logged in.</p> <p>PRE-2: Woodworker views the post list (via "Bài đăng" section in the menu).</p>		
Post-conditions:	<p>POST-1: The new post is added to the post list and is visible to other users.</p> <p>POST-2: Woodworker is redirected to the post list screen after the post is successfully created.</p>		
Normal Flow:	<p>1.0 Create a new post</p> <ol style="list-style-type: none"> 1. Woodworker navigates to the "Bài đăng" section. 2. Woodworker clicks "Thêm bài viết mới" to create a new post. 3. Woodworker fills in the post title and content. 4. Woodworker attaches any images or media to the post (optional). 5. Woodworker submits the post. 6. System saves the post and displays it in the post list. 		
Alternative Flows:	1.1 Missing required content		

	<ol style="list-style-type: none"> 1. If the woodworker submits the post with missing required fields (e.g., title), the system displays a validation error message. 2. Woodworker corrects the content and resubmits.
Exceptions:	<p>1.0.E1 Post save failure</p> <ol style="list-style-type: none"> 1. If the system fails to save the post, an error message is displayed: “Không thể tạo bài viết. Vui lòng thử lại sau.”
Priority:	Medium
Frequency of Use:	Frequent
Business Rules:	BR-70
Other Information:	N/A
Assumptions:	Network connection and backend service are functional.

UC-37: Update post

Use Case ID and Name:	UC-37 – Update post		
Created By:	Luân	Date Created:	12/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker selects an existing post to update.		
Description:	The woodworker updates the content of an existing post, including changes to text, images.		
Preconditions:	PRE-1: Woodworker is logged in. PRE-2: Woodworker has a post that can be edited. PRE-3: Woodworker has access to the post they wish to update (e.g., via post list).		
Post-conditions:	POST-1: The post is updated and the new version is displayed in the post list. POST-2: The system confirms that the post has been updated.		
Normal Flow:	<p>1.0 Update a post</p> <ol style="list-style-type: none"> 1. Woodworker navigates to the "Bài đăng" section. 2. Woodworker selects a post to update. 3. Woodworker edits the title, content, or attaches/removes images. 4. Woodworker submits the changes. 5. System saves the updated post and displays it in the post list. 		
Alternative Flows:	<p>1.1 Missing required content</p> <ol style="list-style-type: none"> 1. If required fields (e.g., title or content) are empty, the system displays a validation error message. 2. Woodworker corrects the issue and resubmits. <p>1.2 Post not found</p>		

	<p>1. If the selected post no longer exists (deleted or invalid), the system displays an error message: “Bài viết không tồn tại.”</p>
Exceptions:	<p>1.0.E1 Post update failure</p> <p>1. If the system fails to save the updated post, an error message is displayed: “Không thể cập nhật bài viết. Vui lòng thử lại sau.”</p>
Priority:	Medium
Frequency of Use:	Occasionally
Business Rules:	BR-70
Other Information:	N/A
Assumptions:	Network connection and backend service are functional.

UC-38: Delete post

Use Case ID and Name:	UC-38 – Delete post		
Created By:	Luân	Date Created:	12/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker selects an existing post and decides to delete it.		
Description:	The woodworker deletes a post that they no longer wish to keep on the platform.		
Preconditions:	<p>PRE-1: Woodworker is logged in.</p> <p>PRE-2: Woodworker has an existing post that can be deleted.</p> <p>PRE-3: Woodworker has the necessary permissions to delete the post (i.e., the post must belong to the woodworker).</p>		
Post-conditions:	<p>POST-1: The post is deleted from the platform.</p> <p>POST-2: The deleted post is no longer visible in the post list.</p> <p>POST-3: A confirmation message is displayed, indicating the post has been deleted.</p>		
Normal Flow:	<p>1.0 Delete a post</p> <ol style="list-style-type: none"> 1. Woodworker navigates to the "Bài đăng" section. 2. Woodworker selects a post they want to delete. 3. Woodworker clicks the delete button on the post. 4. System prompts for confirmation: “Bạn có chắc chắn muốn xóa bài viết này?” 5. Woodworker confirms the deletion. 6. System deletes the post and updates the post list to reflect the change. 7. System displays a success message: “Bài viết đã được xóa thành công.” 		

Alternative Flows:	<p>1.1 Cancel deletion</p> <ol style="list-style-type: none"> If the woodworker cancels the deletion, the post remains unchanged, and the system returns to the post list without deleting it. <p>1.2 Post not found</p> <ol style="list-style-type: none"> If the selected post no longer exists (deleted or invalid), the system displays an error message: "Bài viết không tồn tại".
Exceptions:	<p>1.0.E1 Deletion failure</p> <ol style="list-style-type: none"> If the system fails to delete the post, an error message is displayed: "Không thể xóa bài viết. Vui lòng thử lại sau."
Priority:	Medium
Frequency of Use:	Occasionally
Business Rules:	BR-16, BR-70
Other Information:	N/A
Assumptions:	Network connection and backend service are functional.

UC-39: View design idea list

Use Case ID and Name:	UC-39 – View design idea list		
Created By:	Luân	Date Created:	15/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker navigates to the "Thiết kế" section to view available design ideas.		
Description:	The woodworker accesses the list of available design ideas, either by browsing through predefined categories or searching based on certain criteria.		
Preconditions:	PRE-1: Woodworker is logged in PRE-2: The design idea service is available and active		
Post-conditions:	POST-1: A list of design ideas is displayed to the woodworker POST-2: The woodworker can filter, sort, or search through the design ideas		
Normal Flow:	<p>1.0 View design idea list</p> <ol style="list-style-type: none"> Woodworker clicks "Thiết kế" in the main menu. System retrieves all available design ideas and displays them in a list format. System applies default sorting (by most recent or category). Woodworker can scroll through, filter, or search for specific design ideas. 		
Alternative Flows:	1.1 No design ideas available		

	<p>1. If no design ideas are available, the system displays the message: “Không có thiết kế nào”.</p>
Exceptions:	<p>1.0.E1 Data retrieval error</p> <p>1. If the system fails to load the design ideas, display: “Không thể tải danh sách thiết kế. Vui lòng thử lại”.</p>
Priority:	High (Medium, Low)
Frequency of Use:	Frequently
Business Rules:	BR-70
Other Information:	N/A
Assumptions:	The system successfully displays design ideas, and the woodworker has proper permissions to view them.

UC-40: View design idea detail

Use Case ID and Name:	UC-40 – View design idea detail		
Created By:	Luân	Date Created:	15/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker selects a specific design idea from the list to view more detailed information.		
Description:	Woodworker views detailed information about a specific design idea.		
Preconditions:	PRE-1: Woodworker is logged in. PRE-2: The design idea exists in the system.		
Post-conditions:	POST-1: The detailed information of the selected design idea is displayed to the woodworker.		
Normal Flow:	<p>1.0 View design idea detail</p> <ol style="list-style-type: none"> Woodworker clicks “Thiết kế” in the main menu. System retrieves all available design ideas and displays them in a list format. Woodworker clicks “Xem chi tiết” on a selected design idea from the list. System retrieves and displays the detailed information of the design idea, including images, description, material type, and dimensions. 		
Alternative Flows:	<p>1.1 Design idea not found</p> <ol style="list-style-type: none"> If the selected design idea is deleted or no longer available, the system displays “Không tìm thấy thiết kế này.” 		
Exceptions:	<p>1.0.E1 Data retrieval error</p> <ol style="list-style-type: none"> If the system fails to retrieve the design details, display “Không thể tải chi tiết thiết kế. Vui lòng thử lại.” 		

Priority:	High
Frequency of Use:	Occasional
Business Rules:	BR-70
Other Information:	N/A
Assumptions:	Network connection and backend service are functional.

UC-41: Create design idea

Use Case ID and Name:	UC-41 – Create design idea		
Created By:	Luân	Date Created:	15/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker chooses to create a new design idea.		
Description:	Woodworker creates a new design idea, adding it to the list of available designs for future projects or inspiration.		
Preconditions:	PRE-1: Woodworker is logged in. PRE-2: Woodworker has access to the design creation section.		
Post-conditions:	POST-1: The new design idea is saved in the system and is available in the design idea list. POST-2: Woodworker receives a confirmation that the design idea has been successfully created.		
Normal Flow:	1.0 Create a design idea 1. Woodworker clicks “Thiết kế” in the main menu. 2. System retrieves all available design ideas and displays them in a list format. 3. Woodworker clicks “Tạo thiết kế mới” in the design section. 4. System displays the design creation form. 5. Woodworker fills in the form with details. 6. Woodworker submits the form. 7. System validates the inputs, saves the design idea, and confirms creation.		
Alternative Flows:	1.1 Missing required fields 1. If required fields are empty (e.g., design name), the system highlights the missing fields and prevents submission. 1.2 Invalid file type for images 1. If an image is uploaded in an unsupported format (e.g., non-JPEG or non-PNG), the system displays an error and prompts for a valid file type.		
Exceptions:	1.0.E1 Save failure		

	<ol style="list-style-type: none"> If the system fails to save the design idea, it displays the error message “Không thể tạo thiết kế. Vui lòng thử lại.”
Priority:	High
Frequency of Use:	Occasional
Business Rules:	BR-06, BR-70
Other Information:	N/A
Assumptions:	Network connection and backend service are functional.

UC-42: Update design idea

Use Case ID and Name:	UC-42 – Update design idea		
Created By:	Luân	Date Created:	15/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker selects a design idea and chooses to update its details.		
Description:	Woodworker updates the details of an existing design idea.		
Preconditions:	PRE-1: Woodworker is logged in. PRE-2: The design idea exists and is editable (the logged-in woodworker is the creator).		
Post-conditions:	POST-1: The design idea is updated and saved with the new details. POST-2: The updated design idea is reflected in the design idea list.		
Normal Flow:	1.0 Update a design idea <ol style="list-style-type: none"> Woodworker navigates to the "Thiết kế" section. Woodworker selects the design idea to update. Woodworker clicks “Chỉnh sửa” to open the update form. Woodworker fills in the update form. Woodworker submits the updated design. System validates the changes, saves the updated design idea, and confirms the update. 		
Alternative Flows:	1.1 Missing required fields <ol style="list-style-type: none"> If required fields (e.g., title or description) are missing, the system highlights them and prevents submission. 1.2 Invalid image format <ol style="list-style-type: none"> If an image is uploaded in an unsupported format (e.g., non-JPEG or non-PNG), the system displays an error and prompts for a valid file type. 		
Exceptions:	1.0.E1 Save failure <ol style="list-style-type: none"> If the system fails to save the updated design, it displays an error message: “Không thể cập nhật thiết kế. Vui lòng thử lại.” 		

Priority:	High
Frequency of Use:	Occasional
Business Rules:	BR-06, BR-70
Other Information:	N/A
Assumptions:	Network connection and backend service are functional.

UC-43: Delete design idea

Use Case ID and Name:	UC-43 – Delete design idea		
Created By:	Luân	Date Created:	15/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker selects an existing design idea and chooses to delete it.		
Description:	Woodworker deletes a design idea that is no longer needed or relevant.		
Preconditions:	PRE-1: Woodworker is logged in. PRE-2: The design idea exists and belongs to the logged-in woodworker.		
Post-conditions:	POST-1: The design idea is permanently deleted from the system. POST-2: The deleted design idea is removed from the design list.		
Normal Flow:	1.0 Delete a design idea 1. Woodworker navigates to the "Thiết kế" section. 2. Woodworker selects a design idea to delete. 3. Woodworker clicks the delete icon next to the design idea. 4. System prompts for confirmation: "Bạn có chắc chắn muốn xóa thiết kế này?" 5. Woodworker confirms deletion. 6. System deletes the design idea and updates the design list. 7. System displays "Thiết kế đã được xóa thành công."		
Alternative Flows:	1.1 Cancel deletion 1. At step 4, if the woodworker chooses Hủy, the deletion is aborted and the system returns to the design list. 1.2 Design idea not found 1. If the selected design idea has already been deleted, the system displays "Thiết kế không tồn tại".		
Exceptions:	1.0.E1 Deletion failure 1. If the system fails to delete the design idea, it displays "Không thể xóa thiết kế. Vui lòng thử lại."		
Priority:	High		
Frequency of Use:	Occasional		

Business Rules:	BR-16, BR-69, BR-70
Other Information:	N/A
Assumptions:	Network connection and backend service are functional.

UC-44: View review list

Use Case ID and Name:	UC-44 – View review list		
Created By:	Luân	Date Created:	15/04/2025
Primary Actor:	Woodworker	Secondary Actors:	
Trigger:	Woodworker navigates to the "Đánh giá" section.		
Description:	Woodworker views the list of reviews for their products or services.		
Preconditions:	PRE-1: Woodworker is logged in to their account. PRE-2: Woodworker has reviews associated with their services.		
Post-conditions:	POST-1: The review list is displayed to the woodworker.		
Normal Flow:	1.0 View reviews 1. Woodworker selects "Đánh giá" from the menu. 2. System displays a list of reviews associated with the woodworker's products or services. 3. System allows the woodworker to filter reviews by various criteria (e.g., by rating, by product/service). 4. System allows the woodworker to sort reviews by date, rating, or other parameters. 5. Woodworker can select a review to view more details if needed.		
Alternative Flows:	1.1 No reviews found 1. If no reviews are found, the system displays “Chưa có đánh giá nào” (No reviews available).		
Exceptions:	1.0.E1 No reviews available 1. If the system encounters an error retrieving reviews, it displays “Không thể tải đánh giá. Vui lòng thử lại sau.” (Unable to load reviews. Please try again later).		
Priority:	High		
Frequency of Use:	Frequent		
Business Rules:	BR-70		
Other Information:	N/A		
Assumptions:	Network connection and backend service are functional.		

UC-45: View review detail

Use Case ID and Name:	UC-45: View review detail		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to view a specific review from a customer.		
Description:	This use case allows the woodworker to access and view detailed content of a review left by a customer on a product or service.		
Preconditions:	PRE-01: Woodworker is logged into the platform. PRE-03: The review must exist in the system.		
Post-conditions:	POST-01: The selected review detail is displayed.		
Normal Flow:	1.0 View a review detail 1. Woodworker navigates to the “Reviews” section. 2. Selects a specific review. 3. The system loads and displays the review detail.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 Review not found 1. If the review ID is invalid, system displays an error message.		
Priority:	High (Medium, Low)		
Frequency of Use:	Frequently		
Business Rules:	N/A		
Other Information:	The review includes metadata such as rating, date, and customer name.		
Assumptions:	Reviews are already submitted by customers.		

UC-46: Respond to review

Use Case ID and Name:	UC-46: Respond to review		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to respond to customer feedback.		
Description:	This use case allows the woodworker to reply to a review, providing clarification, thanks, or additional information.		
Preconditions:	PRE-01: Woodworker is authenticated. PRE-03: The review exists and belongs to the woodworker.		
Post-conditions:	POST-01: The response is saved and linked to the review.		

Normal Flow:	<p>1.0 Respond to a review</p> <ol style="list-style-type: none"> 1. Woodworker opens the desired review. 2. Enters a response in the input form. 3. Submits the response. 4. The system saves the response and updates the review thread.
Alternative Flows:	N/A
Exceptions:	<p>1.0.E1 Empty response</p> <ol style="list-style-type: none"> 1. If content is empty or invalid, the system shows a validation error.
Priority:	High (Medium, Low)
Frequency of Use:	Occasionally
Business Rules:	N/A
Other Information:	N/A
Assumptions:	Each review can have one response from the woodworker.

UC-47: Change profile visibility

Use Case ID and Name:	UC-47: Change profile visibility		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to hide or show their profile publicly.		
Description:	This use case allows the woodworker to toggle their profile between public and private visibility.		
Preconditions:	PRE-01: Woodworker is logged in.		
Post-conditions:	POST-01: Profile visibility is updated accordingly in the system.		
Normal Flow:	<p>1.0 Toggle profile visibility</p> <ol style="list-style-type: none"> 1. Woodworker opens “Profile Settings.” 2. Toggles the visibility switch. 3. Clicks “Save”. 4. System updates the visibility status. 		
Alternative Flows:	N/A		
Exceptions:	EXC-03: The system fails to save the setting due to server error.		
Priority:	High (Medium, Low)		
Frequency of Use:	Rare		

Business Rules:	BR-03: Only authenticated woodworkers can change the visibility of their profile, and visibility status must be consistent across all public search listings.
Other Information:	N/A
Assumptions:	Visibility affects whether the profile appears in public search.

UC-48: Upgrade service pack

Use Case ID and Name:	UC-48: Upgrade service pack		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	Woodworker wants to subscribe to a premium service package.		
Description:	This use case describes the process where a woodworker selects and purchases an upgraded service pack to unlock additional features.		
Preconditions:	PRE-01: Woodworker is authenticated. PRE-05: The woodworker's wallet has sufficient balance.		
Post-conditions:	POST-01: The upgraded service pack is activated and reflected in the user's profile.		
Normal Flow:	1.0 Upgrade service pack 1. Woodworker goes to "Service Packs." 2. Selects a new plan. 3. Confirms purchase. 4. System processes payment and applies upgrade.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 Payment failed 1. System shows error if payment does not go through.		
Priority:	High (Medium, Low)		
Frequency of Use:	Occasionally		
Business Rules:	N/A		
Other Information:	N/A		
Assumptions:	Admin-defined service packs are available in the system.		

UC-49: Update woodworker profile

Use Case ID and Name:	UC-49: Update woodworker profile
-----------------------	----------------------------------

Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	Woodworker wants to modify their profile information.		
Description:	This use case allows woodworkers to edit their profile details including bio, skills, warranty policy, and service areas.		
Preconditions:	PRE-01: Woodworker is logged in.		
Post-conditions:	POST-01: Updated profile data is saved and shown in the system.		
Normal Flow:	1.0 Update profile 1. Woodworker navigates to profile page. 2. Edits profile fields. 3. Clicks "Save." 4. System saves and reflects changes.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 Validation failed 1. System notifies user to correct invalid input.		
Priority:	High (Medium, Low)		
Frequency of Use:	Occasionally		
Business Rules:	N/A		
Other Information:	N/A		
Assumptions:	All required fields must be filled in before saving.		

UC-50: View service order history

Use Case ID and Name:	UC-50: View service order history		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to see a list of past service orders.		
Description:	This use case allows a woodworker to view a historical list of received service orders.		
Preconditions:	PRE-01: Woodworker is authenticated.		
Post-conditions:	POST-01: List of past service orders is displayed.		
Normal Flow:	1.0 View service history 1. Woodworker goes to "Orders." 2. Clicks on "Service Order History."		

	3. System displays past orders.
Alternative Flows:	N/A
Exceptions:	1.0.E1 Data error 1. If records are missing, system shows "No orders found."
Priority:	High (Medium, Low)
Frequency of Use:	Frequently
Business Rules:	N/A
Other Information:	N/A
Assumptions:	Orders are stored and associated with the woodworker's ID.

UC-51: View service order detail

Use Case ID and Name:	UC-51: View service order detail		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to view the full detail of a service order.		
Description:	This use case allows the woodworker to access all information related to a specific service order including customer details and service specifications.		
Preconditions:	PRE-01: Woodworker is authenticated. PRE-03: The service order exists.		
Post-conditions:	POST-01: The service order detail is displayed to the woodworker.		
Normal Flow:	1.0 View service order detail 1. Woodworker goes to the "Orders" section. 2. Selects a specific service order. 3. System retrieves and displays all service order data.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 Order not found 1. System shows an error message if the order ID is invalid or deleted.		
Priority:	High (Medium, Low)		
Frequency of Use:	Frequently		
Business Rules:	N/A		
Other Information:	N/A		
Assumptions:	The woodworker only sees orders assigned to them.		

UC-52: View guarantee order history

Use Case ID and Name:	UC-52: View guarantee order history		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to check the list of guarantee orders they've received.		
Description:	This use case allows the woodworker to view a history of all past guarantee (warranty) service orders.		
Preconditions:	PRE-01: Woodworker is logged in.		
Post-conditions:	POST-01: A list of past guarantee orders is displayed.		
Normal Flow:	1.0 View guarantee order history 1. Woodworker accesses the "Guarantee Orders" section. 2. Clicks on the "History" tab. 3. System displays the list of previously handled guarantee orders.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 No orders found 1. System displays "No guarantee orders available."		
Priority:	High (Medium, Low)		
Frequency of Use:	Occasionally		
Business Rules:	N/A		
Other Information:	N/A		
Assumptions:	Guarantee orders are recorded and stored by system.		

UC-53: View guarantee order detail

Use Case ID and Name:	UC-53: View guarantee order detail		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to view full information of a specific guarantee order.		
Description:	This use case allows the woodworker to open and review all details of a selected guarantee service order.		
Preconditions:	PRE-01: Woodworker is logged in. PRE-03: The selected guarantee order exists.		

Post-conditions:	POST-01: The guarantee order detail is displayed.
Normal Flow:	1.0 View guarantee order detail 1. Woodworker opens “Guarantee Orders.” 2. Selects a specific order. 3. System loads and displays the order detail.
Alternative Flows:	N/A
Exceptions:	1.0.E1 Invalid order 1. System shows error if the order ID is invalid or deleted.
Priority:	High (Medium, Low)
Frequency of Use:	Frequently
Business Rules:	N/A
Other Information:	N/A
Assumptions:	The woodworker only sees orders assigned to them.

UC-54: Update consultant appointment

Use Case ID and Name:	UC-54: Update consultant appointment		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to reschedule or update a consultation appointment.		
Description:	This use case allows the woodworker to update the time, note, or status of an upcoming consultation session with a customer.		
Preconditions:	PRE-01: Woodworker is logged in. PRE-03: A consultation appointment exists.		
Post-conditions:	POST-01: The consultation appointment is updated in the system.		
Normal Flow:	1.0 Update consultant appointment 1. Woodworker opens “Consultation Schedule.” 2. Selects the appointment to update. 3. Edits date/time or status. 4. System saves and confirms the changes.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 Invalid date/time 1. If selected date is in the past, system blocks the change.		
Priority:	High (Medium, Low)		

Frequency of Use:	Frequently
Business Rules:	N/A
Other Information:	N/A
Assumptions:	Appointments are stored with timestamps and can be modified before the meeting.

UC-55: Update service order contract

Use Case ID and Name:	UC-55: Update service order contract		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to update terms or conditions in a service contract.		
Description:	This use case enables the woodworker to update contract details tied to a service order (e.g. pricing, deliverables, schedule).		
Preconditions:	PRE-01: Woodworker is logged in. PRE-03: The service order contract exists and is editable.		
Post-conditions:	POST-01: Contract details are updated and reflected in the system.		
Normal Flow:	1.0 Update service order contract 1. Woodworker accesses "Service Contracts." 2. Selects a contract to update. 3. Modifies fields (e.g., terms, pricing). 4. System saves the changes and updates the contract view.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 Contract locked 1. If the contract is already signed or finalized, the system blocks changes.		
Priority:	High (Medium, Low)		
Frequency of Use:	Occasionally		
Business Rules:	N/A		
Other Information:	N/A		
Assumptions:	Only editable contracts can be modified by the assigned woodworker.		

UC-56: Update service order product design

Use Case ID and Name:	UC-56: Update service order product design
-----------------------	--

Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to edit or adjust the product design details for a service order.		
Description:	This use case allows the woodworker to update technical specifications, design sketches, or notes related to a product's design.		
Preconditions:	PRE-01: Woodworker is logged in. PRE-03: Service order with design exists.		
Post-conditions:	POST-01: Updated design information is saved to the order.		
Normal Flow:	1.0 Update service order design 1. Woodworker accesses a service order. 2. Clicks "Edit Design." 3. Updates content (e.g., drawing, material specs). 4. System saves and reflects the updates.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 Missing design section 1. System shows error if design field is empty.		
Priority:	High (Medium, Low)		
Frequency of Use:	Occasionally		
Business Rules:	N/A		
Other Information:	N/A		
Assumptions:	System supports design upload or text input.		

UC-57: Update service order finish product image

Use Case ID and Name:	UC-57: Update service order finish product image		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to upload or change the image of the completed product.		
Description:	This use case allows the woodworker to upload final images of the completed product tied to the service.		
Preconditions:	PRE-01: Woodworker is authenticated. PRE-03: Service status is "Completed".		

Post-conditions:	POST-01: Image is linked to the corresponding service order.
Normal Flow:	1.0 Upload finished product image 1. Woodworker opens a completed service order. 2. Clicks “Upload Image.” 3. Selects image and submits. 4. System saves the image.
Alternative Flows:	N/A
Exceptions:	1.0.E1 Upload failed 1. File format or size invalid, system shows error.
Priority:	High (Medium, Low)
Frequency of Use:	Occasionally
Business Rules:	N/A
Other Information:	N/A
Assumptions:	Image upload system is properly configured.

UC-58: Update service order quotation

Use Case ID and Name:	UC-58: Update service order quotation		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to revise the service quotation (price, duration, etc).		
Description:	This use case allows the woodworker to update quote details before the service starts.		
Preconditions:	PRE-01: Woodworker is authenticated. PRE-03: The quotation is not yet confirmed by customer.		
Post-conditions:	POST-01: Quotation is updated and reflected in the system.		
Normal Flow:	1.0 Update service quotation 1. Woodworker opens service request. 2. Clicks “Edit Quotation.” 3. Updates price, timeline or terms. 4. Saves changes.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 Quotation locked 1. If customer already approved quote, system prevents editing.		
Priority:	High (Medium, Low)		

Frequency of Use:	Occasionally
Business Rules:	N/A
Other Information:	N/A
Assumptions:	Quotation is editable before customer confirms.

UC-59: Update guarantee order quotation

Use Case ID and Name:	UC-59: Update guarantee order quotation		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker needs to add/edit the quotation for a warranty/guarantee repair.		
Description:	This use case allows the woodworker to update cost estimation for guarantee services.		
Preconditions:	PRE-01: Woodworker is logged in. PRE-03: Guarantee order is assigned.		
Post-conditions:	POST-01: The quotation is saved and linked to the guarantee order.		
Normal Flow:	1.0 Update guarantee order quotation 1. Woodworker selects a guarantee order. 2. Clicks “Update Quotation.” 3. Enters or modifies price and terms. 4. Confirms and saves the update.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 Edit denied 1. If customer already declined or accepted a previous quote.		
Priority:	High (Medium, Low)		
Frequency of Use:	Occasionally		
Business Rules:	N/A		
Other Information:	N/A		
Assumptions:	Guarantee quotations follow separate workflow from regular orders.		

UC-60: View service list

Use Case ID and Name:	UC-60: View service list
-----------------------	--------------------------

Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to view a full list of their active services.		
Description:	This use case allows the woodworker to see all service orders they are assigned to.		
Preconditions:	PRE-01: Woodworker is authenticated.		
Post-conditions:	POST-01: The list of current services is shown.		
Normal Flow:	1.0 View service list 1. Woodworker opens the “Service List” section. 2. System loads all active orders. 3. User can click any service to view details.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 No services found 1. System shows “No current services” message.		
Priority:	High (Medium, Low)		
Frequency of Use:	Frequently		
Business Rules:	N/A		
Other Information:	N/A		
Assumptions:	All assigned services are retrieved from database.		

UC-61: Update service

Use Case ID and Name:	UC-61: Update service		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to update general service information.		
Description:	This use case enables the woodworker to update service content (name, category, images, price).		
Preconditions:	PRE-01: Woodworker is authenticated.		
Post-conditions:	POST-01: The service information is updated.		
Normal Flow:	1.0 Update service 1. Woodworker goes to “My Services.” 2. Selects a service to edit.		

	3. Modifies name, price, or description. 4. Saves changes.
Alternative Flows:	N/A
Exceptions:	1.0.E1 Save failed 1. Server or validation error prevents update.
Priority:	High (Medium, Low)
Frequency of Use:	Occasionally
Business Rules:	N/A
Other Information:	N/A
Assumptions:	System validates and stores edited service fields.

UC-62: View complaint list

Use Case ID and Name:	UC-62: View complaint list		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to review all complaints related to their services.		
Description:	This use case allows the woodworker to see a list of complaints made by customers.		
Preconditions:	PRE-01: Woodworker is authenticated.		
Post-conditions:	POST-01: A list of complaints is shown.		
Normal Flow:	1.0 View complaint list 1. Woodworker goes to “Complaints.” 2. System loads list of complaints linked to their services.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 No complaints 1. System shows “No complaints available.”		
Priority:	High (Medium, Low)		
Frequency of Use:	Occasionally		
Business Rules:	N/A		
Other Information:	N/A		
Assumptions:	Complaints are linked to services by woodworker ID.		

UC-63: View complaint detail

Use Case ID and Name:	UC-63: View complaint detail		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to view specific complaint content.		
Description:	This use case allows the woodworker to read full details of a customer's complaint.		
Preconditions:	PRE-01: Woodworker is logged in. PRE-03: A complaint is assigned to their service.		
Post-conditions:	POST-01: Complaint detail is displayed.		
Normal Flow:	1.0 View complaint detail 1. Woodworker selects a complaint from list. 2. System shows full complaint info.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 Complaint not found 1. System shows error if complaint is missing.		
Priority:	High (Medium, Low)		
Frequency of Use:	Occasionally		
Business Rules:	N/A		
Other Information:	N/A		
Assumptions:	Each complaint is stored per service order.		

UC-64: Respond complaint

Use Case ID and Name:			
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	N/A
Trigger:	The woodworker wants to respond to a complaint issued by a customer.		
Description:	This use case allows the woodworker to send a reply or explanation to the complaint.		
Preconditions:	PRE-01: Woodworker is logged in. PRE-03: Complaint is still active.		
Post-conditions:	POST-01: Response is saved and linked to the complaint.		

Normal Flow:	1.0 Respond to complaint 1. Woodworker opens a complaint. 2. Types response and submits. 3. System saves reply.
Alternative Flows:	N/A
Exceptions:	1.0.E1 Response failed 1. Validation or server error occurs.
Priority:	High (Medium, Low)
Frequency of Use:	Occasionally
Business Rules:	N/A
Other Information:	N/A
Assumptions:	Each complaint allows one response from woodworker.

UC-65: View transaction history

Use Case ID and Name:	UC-65: View transaction history		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	Customer
Trigger:	The user wants to view their wallet transaction history.		
Description:	This use case allows both the woodworker and customer to see past wallet activity, including income and payments.		
Preconditions:	PRE-01: User is authenticated.		
Post-conditions:	POST-01: System displays a chronological list of transactions.		
Normal Flow:	1.0 View transaction history 1. User opens the “Wallet” or “Finance” section. 2. Clicks “Transaction History.” 3. System retrieves and displays all transaction entries.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 No transaction data 1. If no records are found, system shows “No transactions available.”		
Priority:	High (Medium, Low)		
Frequency of Use:	Occasionally		
Business Rules:	N/A		
Other Information:	History includes timestamps, amount, and transaction type.		

Assumptions:	All transactions are securely linked to user ID in the database.
--------------	--

UC-66: View transaction detail

Use Case ID and Name:	UC-66: View transaction detail		
Created By:	Nam	Date Created:	23/04/2025
Primary Actor:	Woodworker	Secondary Actors:	Customer
Trigger:	The user clicks on a transaction to see more detail.		
Description:	This use case allows users to access full details for any individual transaction (e.g., type, method, receiver, date, and status).		
Preconditions:	PRE-01: User is logged in. PRE-03: Transaction exists and is linked to user.		
Post-conditions:	POST-01: System displays transaction detail info.		
Normal Flow:	1.0 View transaction detail 1. User opens “Transaction History.” 2. Clicks on a specific entry. 3. System displays detailed transaction data.		
Alternative Flows:	N/A		
Exceptions:	1.0.E1 Invalid transaction ID 1. System shows error if transaction is missing or unauthorized.		
Priority:	High (Medium, Low)		
Frequency of Use:	Occasionally		
Business Rules:	N/A		
Other Information:	N/A		
Assumptions:	Users can only view details of transactions that belong to them.		

UC-67: Deposit money

Use Case ID and Name:	UC-67: Deposit money		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Woodworker + Customer	Secondary Actors:	N/A
Trigger:	The user decides to deposit money into their account.		

Description:	This use case describes the process by which a woodworker or user deposits money into their account. It includes selecting the deposit method, entering the amount, and confirming the transaction.
Preconditions:	The user is logged into their account. The user has access to a deposit method (e.g., bank account). The account has been verified.
Post-conditions:	The deposit amount is successfully added to the user's balance. A confirmation message or receipt is generated.
Normal Flow:	<ol style="list-style-type: none"> 1. The user logs into his account. 2. The User navigates to "Quản lý thông tin" 3. The User navigates to "Ví" 4. The User navigates to "Gửi tiền" 5. The user enters the amount to deposit. 6. The User confirms the amount to deposit. 7. The system authenticates the deposit amount. 8. The user selects the deposit method. 9. The user fills in payment account information. 10. The system authenticates the deposit method. 11. The system processes the deposit order and updates the account balance.
Alternative Flows:	<p>Invalid Deposit Amount:</p> <ul style="list-style-type: none"> • If the user enters an invalid amount (e.g., negative value, exceeding account limits), the system displays an error message and prompts the user to enter a valid amount. <p>Failed Deposit Method:</p> <ul style="list-style-type: none"> • If the selected deposit method fails (e.g., bank transfer error), the system will inform the user and prompt them to try another method or contact support.
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> • If the deposit cannot be processed due to network issues, the system displays an error message and prompts the user to try again later. <p>Insufficient Funds Deposit Method:</p> <ul style="list-style-type: none"> • If the deposit method does not have enough funds, the system displays an error and prompts the user to select another method or provide additional funds.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-72
Other Information:	The deposit will be reflected in the user's account balance immediately after successful processing.
Assumptions:	The user has a valid and active payment method available for use. The user is authorized to perform deposit transactions within the platform.

UC-68: Request withdrawal

Use Case ID and Name:	UC-68: Request Withdrawal		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Woodworker + Customer	Secondary Actors:	N/A
Trigger:	The user decides to request a withdrawal from their account.		
Description:	This use case describes the process by which a woodworker or user requests a withdrawal from their account. It includes selecting the withdrawal amount, choosing a withdrawal method, and confirming the transaction.		
Preconditions:	<p>The user is logged into their account.</p> <p>The user has sufficient balance to withdraw.</p> <p>The user has access to a valid withdrawal method (e.g., bank account).</p>		
Post-conditions:	<p>The requested withdrawal amount is successfully processed.</p> <p>The user's account balance is updated.</p> <p>A confirmation message or receipt is generated.</p>		
Normal Flow:	<ol style="list-style-type: none"> 1. The user logs into their account. 2. The user navigates to "Quản lý thông tin" (Manage Information). 3. The user navigates to "Ví" (Wallet). 4. The user navigates to "Rút tiền" (Withdraw Money). 5. The user enters the amount to withdraw. 6. The user confirms the withdrawal amount. 7. The system authenticates the withdrawal amount and ensures sufficient funds are available. 8. The user selects the withdrawal method (e.g., bank transfer). 9. The user fills in the required payment account details. 10. The system authenticates the withdrawal method. 11. The system processes the withdrawal request and updates the account balance. 12. A confirmation message or receipt is displayed to the user. 		
Alternative Flows:	<p>Invalid Withdrawal Amount:</p> <ul style="list-style-type: none"> • If the user enters an invalid amount (e.g., negative value, exceeds available balance), the system displays an error message and prompts the user to enter a valid amount. <p>Failed Withdrawal Method:</p> <ul style="list-style-type: none"> • If the selected withdrawal method fails (e.g., bank transfer error), the system informs the user and prompts them to try another method or contact support. 		
Exceptions:	Network Error:		

	<ul style="list-style-type: none"> If the withdrawal cannot be processed due to network issues, the system displays an error message and prompts the user to try again later. <p>Insufficient Funds:</p> <ul style="list-style-type: none"> If the user does not have enough funds for the withdrawal, the system displays an error message and prompts the user to adjust the withdrawal amount or check their balance.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-73
Other Information:	<p>The withdrawal will be reflected in the user's account balance after successful processing.</p> <p>A transaction fee may apply for certain withdrawal methods.</p>
Assumptions:	<ul style="list-style-type: none"> The user has a valid and active payment method available for withdrawals. The user is authorized to perform withdrawal transactions within the platform.

UC-69: View woodworker performance list

Use Case ID and Name:	UC-69: View Woodworker Performance List		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Moderator	Secondary Actors:	N/A
Trigger:	The moderator requests to view the list of woodworker performance.		
Description:	This use case describes the process by which a moderator views the performance list of all woodworkers. The moderator can see performance metrics, activity summaries, and ranking.		
Preconditions:	<ul style="list-style-type: none"> The moderator is logged into the system. The moderator has the required permissions to view performance data. 		
Post-conditions:	The system displays the performance list of all woodworkers, including relevant data such as performance score, task completion rate, and ranking.		
Normal Flow:	<ol style="list-style-type: none"> The moderator logs into their account. The moderator navigates to the "Hiệu Suất" section. The system displays the list of all woodworkers with their performance data. 		

	<p>4. The moderator can filter or sort the list based on various criteria (e.g., performance score, date range).</p>
Alternative Flows:	<p>Network Error:</p> <ul style="list-style-type: none"> If the system cannot retrieve performance data due to network issues, it will display an error message and prompt the moderator to try again later.
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> If the system cannot retrieve performance data due to network issues, it will display an error message and prompt the moderator to try again later.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-74, BR-75
Other Information:	The moderator can export the list of performance data for further analysis.
Assumptions:	The moderator has access to all necessary data to evaluate woodworker performance.

UC-70: View woodworker performance detail

Use Case ID and Name:	UC-70: View woodworker performance detail		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Moderator	Secondary Actors:	N/A
Trigger:	The moderator selects a specific woodworker to view detailed performance information.		
Description:	This use case describes the process by which a moderator views the detailed performance of a single woodworker. The moderator can see individual task performance, feedback, and history.		
Preconditions:	<p>The moderator is logged into their account.</p> <p>The moderator has the necessary permissions to access detailed performance data.</p>		
Post-conditions:	The system displays detailed performance metrics of a selected woodworker, such as completed tasks, time spent, and feedback.		
Normal Flow:	<ol style="list-style-type: none"> The moderator logs into their account. The moderator navigates to the "Hiệu Suất" section. The system displays the list of all woodworkers with their performance data. 		

	<ol style="list-style-type: none"> 4. The moderator selects a specific woodworker from the performance list. 5. The system retrieves and displays the detailed performance data of the selected woodworker. 6. The moderator can view detailed metrics and historical performance.
Alternative Flows:	No Detailed Data Available: <ul style="list-style-type: none"> • If no detailed performance data is available for the selected woodworker, the system will display a message indicating the lack of data.
Exceptions:	Network Error: <ul style="list-style-type: none"> • If the system cannot retrieve detailed performance data due to network issues, an error message is displayed.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-76
Other Information:	The moderator can download or export the detailed performance data.
Assumptions:	The woodworker's performance data has been properly recorded and updated.

UC-71: Send performance email

Use Case ID and Name:	UC-71: Send Performance Email		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Moderator	Secondary Actors:	Woodworker (recipient of the performance email)
Trigger:	The moderator triggers the action to send a performance email to a woodworker.		
Description:	This use case describes the process by which the moderator sends an email detailing the performance of a woodworker. The email may include feedback, performance metrics, and suggestions for improvement.		
Preconditions:	The moderator is logged into their account. The moderator has the necessary permissions to send performance emails.		
Post-conditions:	The woodworker receives an email detailing their performance metrics and feedback.		
Normal Flow:	<ol style="list-style-type: none"> 1. The moderator logs into their account. 2. The moderator navigates to the "Hiệu suất" section. 3. The moderator selects a woodworker from the performance list. 4. The moderator clicks the "Gửi email hiệu suất" button. 		

	<p>5. The system automatically generates and sends an email with the performance data to the selected woodworker.</p>
Alternative Flows:	<p>Email Sending Failure:</p> <ul style="list-style-type: none"> If the system fails to send the email (e.g., email server issue), the system displays an error message and prompts the moderator to try again later.
Exceptions:	<p>Invalid Email Address:</p> <ul style="list-style-type: none"> If the email address is invalid or unreachable, the system displays an error message and prompts the moderator to verify the email address.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-77, BR-78
Other Information:	The system can send automated performance emails at scheduled intervals, or the moderator can send them manually.
Assumptions:	<p>The woodworker's email address is available and valid in the system.</p> <p>The email system is functioning properly.</p>

UC-72: View review list

Use Case ID and Name:	UC-72: View review list		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Staff	Secondary Actors:	N/A
Trigger:	The staff member requests to view the list of customer reviews.		
Description:	This use case describes the process by which a staff member views the list of reviews submitted by customers. The list includes review ratings, comments, and review dates.		
Preconditions:	<p>The staff member is logged into their account.</p> <p>The staff member has permission to view customer reviews.</p>		
Post-conditions:	The system displays a list of reviews with ratings and comments.		
Normal Flow:	<ol style="list-style-type: none"> The staff member logs into their account. The staff member navigates to the "Đánh Giá Của Khách Hàng" section. The system displays a list of reviews, showing ratings, comments, and submission dates. 		
Alternative Flows:	No Reviews Available:		

	<ul style="list-style-type: none"> If no reviews are available, the system will notify the staff member with a message.
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> If the reviews cannot be retrieved due to network issues, an error message is displayed.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-79
Other Information:	The staff member can filter or sort the list based on specific criteria like rating or date.
Assumptions:	Reviews are recorded and updated correctly in the system.

UC-73: View review detail

Use Case ID and Name:	UC-73: View Review Detail		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Staff	Secondary Actors:	N/A
Trigger:	The staff member requests to view the details of a specific customer review.		
Description:	This use case describes the process by which a staff member views the full details of a specific review submitted by a customer. It includes review content, ratings, and customer information.		
Preconditions:	<p>The staff member is logged into their account.</p> <p>The staff member has permission to view review details.</p>		
Post-conditions:	The system displays the full details of the selected review.		
Normal Flow:	<ol style="list-style-type: none"> The staff member logs into their account. The staff member navigates to the "Đánh Giá Của Khách Hàng" section. The staff member selects a review from the review list. The system displays the full details of the selected review, including rating, comments, and customer information. 		
Alternative Flows:	<p>No Detailed Data:</p> <ul style="list-style-type: none"> If no detailed review data is available, the system will notify the staff member. 		
Exceptions:	Network Error:		

	<ul style="list-style-type: none"> If the review details cannot be retrieved due to network issues, an error message is displayed.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-79
Other Information:	The staff member may be able to export the review details if needed.
Assumptions:	Reviews are recorded and updated correctly in the system.

UC-74: Review customer review

Use Case ID and Name:	UC-74: Review Customer Review		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Staff	Secondary Actors:	Customer (who submitted the review)
Trigger:	The staff member reviews a customer's submitted review.		
Description:	This use case allows the staff member to review a customer's submitted review, providing feedback, and possibly moderating the review content.		
Preconditions:	<p>The staff member is logged into their account.</p> <p>The staff member has permission to review and moderate customer reviews.</p>		
Post-conditions:	The staff member provides feedback or approves the review for publication.		
Normal Flow:	<ol style="list-style-type: none"> The staff member logs into their account. The staff member navigates to the "Đánh Giá Của Khách Hàng" section. The staff member selects a customer review for review. The system displays the full details of the customer's review. The staff member reviews the content and provides feedback, either approving or flagging the review for further review. 		
Alternative Flows:	<p>Review Flagged for Inappropriate Content:</p> <ul style="list-style-type: none"> If the review contains inappropriate content, the system will allow the staff member to flag it for further review. 		
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> If the review data cannot be fetched, the system will show an error message. 		
Priority:	High		
Frequency of Use:	Frequent		

Business Rules:	BR-80
Other Information:	The review may be flagged for manual review by a higher authority.
Assumptions:	Reviews are recorded and updated properly in the system.

UC-75: View response list

Use Case ID and Name:	UC-75: View response list		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Staff	Secondary Actors:	N/a
Trigger:	The staff member requests to view the list of responses provided to customer reviews.		
Description:	This use case describes the process by which a staff member views a list of responses made by customer reviews.		
Preconditions:	<p>The staff member is logged into their account.</p> <p>The staff member has permission to view responses.</p>		
Post-conditions:	The system displays a list of responses made by customer reviews.		
Normal Flow:	<ol style="list-style-type: none"> 1. The employee logs into their account. 2. The employee navigates to the "Woodshop Feedback" section. 3. The system displays the list of responses to customer reviews. 		
Alternative Flows:	<p>No Responses Available:</p> <ul style="list-style-type: none"> • If no responses are available, the system displays a message indicating no responses have been made. 		
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> • If responses cannot be retrieved due to network issues, an error message is displayed. 		
Priority:	High		
Frequency of Use:	Frequent		
Business Rules:	BR-79		
Other Information:	The staff member can filter or sort responses based on review date, rating, etc.		
Assumptions:	Responses are properly recorded and updated in the system.		

UC-76: View response detail

Use Case ID and Name:	UC-76: View response detail		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Staff	Secondary Actors:	N/A
Trigger:	The staff member selects a response to view its full content.		
Description:	This use case allows the staff member to view detailed information of a response made by a customer review.		
Preconditions:	<p>The staff member is logged into their account.</p> <p>The staff member has permission to view response details.</p>		
Post-conditions:	The system displays the full content of the selected response.		
Normal Flow:	<ol style="list-style-type: none"> 1. The employee logs into their account. 2. The employee navigates to the "Woodshop Feedback" section. 3. The system displays the list of responses to customer reviews. 4. The system displays the detailed content of the response. 		
Alternative Flows:	<p>No Detailed Response Data:</p> <ul style="list-style-type: none"> • If no detailed response data is available, the system will notify the staff member. 		
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> • If response details cannot be retrieved, an error message is displayed. 		
Priority:	High		
Frequency of Use:	Frequent		
Business Rules:	BR-81		
Other Information:	The staff member may be able to export the response details if needed.		
Assumptions:	Responses are correctly recorded in the system.		

UC-77: Review woodworker response

Use Case ID and Name:	UC-77: Review Woodworker Response		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Staff	Secondary Actors:	Woodworker
Trigger:	The staff member reviews a response made by a woodworker.		
Description:	This use case allows the staff member to review and moderate the response submitted by a woodworker to customer reviews.		

Preconditions:	The staff member is logged into their account. The staff member has permission to review woodworker responses.
Post-conditions:	The staff member provides feedback or approves/rejects the response.
Normal Flow:	<ol style="list-style-type: none"> 1. The staff member logs into their account. 2. The staff member navigates to the "Phản hồi của xưởng mộc" section. 3. The system displays the full details of the response. 4. The staff member reviews the response and either approves or flags it for further review.
Alternative Flows:	<p>Response Flagged for Review:</p> <ul style="list-style-type: none"> • If the response is inappropriate or violates policies, the staff member can flag it.
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> • If the response data cannot be retrieved, an error message is displayed.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-82
Other Information:	The staff member can provide feedback on the response.
Assumptions:	Responses are recorded correctly in the system.

UC-78: View complaint list

Use Case ID and Name:	UC-78: View complaint list		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Staff	Secondary Actors:	N/A
Trigger:	The staff member requests to view the list of customer complaints.		
Description:	This use case allows the staff member to view a list of complaints submitted by customers.		
Preconditions:	<p>The staff member is logged into their account.</p> <p>The staff member has permission to view complaints.</p>		
Post-conditions:	The system displays a list of customer complaints.		

Normal Flow:	<ol style="list-style-type: none"> 1. The staff member logs into their account. 2. The staff member navigates to the "Khiếu Nại" section. 3. The system displays the list of complaints.
Alternative Flows:	<p>No Complaints Available:</p> <ul style="list-style-type: none"> • If no complaints are available, the system will notify the staff member.
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> • If complaints cannot be retrieved due to network issues, an error message is displayed.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-83
Other Information:	The staff member can filter or sort complaints.
Assumptions:	Complaints are correctly recorded in the system.

UC-79: View complaint detail

Use Case ID and Name:	UC-79: View Complaint Detail		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Staff	Secondary Actors:	N/A
Trigger:	The staff member selects a complaint to view its full details.		
Description:	This use case allows the staff member to view detailed information regarding a specific customer complaint.		
Preconditions:	The staff member is logged into their account. The staff member has permission to view complaint details.		
Post-conditions:	The system displays the full details of the selected complaint.		
Normal Flow:	<ol style="list-style-type: none"> 1. The staff member logs into their account. 2. The staff member navigates to the "Khiếu Nại" section. 3. The staff member selects a complaint from the list. 4. The system displays the details of the selected complaint. 		
Alternative Flows:	<p>No Detailed Data Available:</p> <ul style="list-style-type: none"> • If no detailed data is available, the system will notify the staff member. 		
Exceptions:	Network Error:		

	<ul style="list-style-type: none"> If the system cannot retrieve the complaint details, an error message is displayed.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-84
Other Information:	The staff member may export the complaint details if needed.
Assumptions:	Complaints are properly recorded and updated in the system.

UC-80: Process complaint

Use Case ID and Name:	UC-80: Process complaint		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Staff	Secondary Actors:	N/A
Trigger:	The staff member processes a customer complaint by taking appropriate action.		
Description:	This use case describes how the staff member processes a customer complaint, including reviewing the complaint and determining the next steps (e.g., resolving the issue, escalating to management).		
Preconditions:	<p>The staff member is logged into their account.</p> <p>The staff member has permission to process complaints.</p>		
Post-conditions:	The system updates the status of the complaint (e.g., resolved, under review, escalated).		
Normal Flow:	<ol style="list-style-type: none"> The staff member logs into their account. The staff member navigates to the "Khiếu Nại" section. The staff member selects a complaint to process. The staff member reviews the complaint and determines an action (e.g., resolve, escalate). The system updates the complaint status based on the staff member's actions. 		
Alternative Flows:	<p>Escalation Needed:</p> <ul style="list-style-type: none"> If the complaint needs to be escalated to a higher authority, the system allows the staff member to forward the complaint. 		
Exceptions:	Network Error:		

	<ul style="list-style-type: none"> If the system cannot update the complaint status, an error message is displayed.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-85
Other Information:	The staff member can add notes or comments when processing the complaint.
Assumptions:	Complaints are correctly recorded in the system.

UC-81: View transaction history

Use Case ID and Name:	UC-81: View transaction history		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Admin	Secondary Actors:	N/A
Trigger:	The admin requests to view the list of transaction history.		
Description:	This use case describes the process by which an admin views the history of all transactions. It includes transaction details such as amount, date, and type.		
Preconditions:	<p>The admin is logged into the system.</p> <p>The admin has permission to view transaction history.</p>		
Post-conditions:	The system displays the list of transactions with their details.		
Normal Flow:	<ol style="list-style-type: none"> The admin logs into the system. The admin navigates to the "Giao dịch" section. The system displays the list of transactions, including amount, type, and date. 		
Alternative Flows:	<p>No Transactions Available:</p> <ul style="list-style-type: none"> If no transactions are available, the system notifies the admin with a message. 		
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> If the transaction data cannot be retrieved due to network issues, an error message is displayed. 		
Priority:	High		
Frequency of Use:	Frequent		
Business Rules:	BR-86		

Other Information:	The admin may be able to filter or sort the transaction list by amount, type, or date.
Assumptions:	Transaction data is accurately recorded in the system.

UC-82: View transaction detail

Use Case ID and Name:	UC-82: View transaction detail		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Admin	Secondary Actors:	N/A
Trigger:	The admin selects a specific transaction to view its full details.		
Description:	This use case allows the admin to view detailed information for a specific transaction, including the involved parties and transaction status.		
Preconditions:	<p>The admin is logged into the system.</p> <p>The admin has permission to view transaction details.</p>		
Post-conditions:	The system displays the full details of the selected transaction.		
Normal Flow:	<ol style="list-style-type: none"> 1. The admin logs into the system. 2. The admin navigates to the "Giao dịch" section. 3. The system displays the list of transactions 4. The admin selects a transaction to view. 5. The system displays the detailed information of the selected transaction. 		
Alternative Flows:	<p>No Detailed Data:</p> <ul style="list-style-type: none"> • If no detailed data is available, the system notifies the admin. 		
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> • If the transaction details cannot be retrieved due to network issues, an error message is displayed. 		
Priority:	High		
Frequency of Use:	Frequent		
Business Rules:	BR-87		
Other Information:	The admin may be able to export transaction details.		
Assumptions:	Transaction data is stored and maintained correctly in the system.		

UC-83: View transaction and revenue charts

Use Case ID and Name:	UC-83: View transaction and revenue charts		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Admin	Secondary Actors:	N/A
Trigger:	The admin requests to view transaction and revenue charts.		
Description:	This use case allows the admin to view graphical representations of transaction data and revenue trends over a specified time period.		
Preconditions:	<p>The admin is logged into the system.</p> <p>The admin has permission to access transaction and revenue data.</p>		
Post-conditions:	The system displays charts showing transaction volume and revenue trends		
Normal Flow:	<ol style="list-style-type: none"> 1. The admin logs into the system. 2. The admin navigates to the "Doanh thu" section. 3. The system displays graphical charts showing transaction and revenue data. 		
Alternative Flows:	<p>No Data Available:</p> <ul style="list-style-type: none"> • If no transaction or revenue data is available, the system displays a message indicating this. 		
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> • If the chart data cannot be retrieved due to network issues, an error message is displayed. 		
Priority:	High		
Frequency of Use:	Frequent		
Business Rules:	BR-88		
Other Information:	The admin can filter the data by different time periods or categories.		
Assumptions:	Transaction and revenue data are up-to-date and accurately recorded.		

UC-84: View woodworker registration list

Use Case ID and Name:	UC-84: View woodworker registration list		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Admin	Secondary Actors:	N/A
Trigger:	The admin requests to view the list of woodworker registrations.		

Description:	This use case allows the admin to view the list of woodworkers who have registered on the platform, including their registration details.
Preconditions:	The admin is logged into the system. The admin has permission to view woodworker registration data.
Post-conditions:	The system displays the list of woodworker registrations.
Normal Flow:	<ol style="list-style-type: none"> 1. The admin logs into the system. 2. The admin navigates to the "Đơn đăng ký xưởng" section. 3. The system displays the list of registered woodworkers, including their registration details.
Alternative Flows:	No Woodworker Registrations: <ul style="list-style-type: none"> • If no woodworkers are registered, the system notifies the admin.
Exceptions:	Network Error: <ul style="list-style-type: none"> • If the registration data cannot be retrieved due to network issues, an error message is displayed.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-89
Other Information:	The admin may filter or sort the list based on registration date or woodworker name.
Assumptions:	Woodworker registration data is accurately recorded and updated in the system.

UC-85: View woodworker registration detail

Use Case ID and Name:	UC-85: View woodworker registration detail		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Admin	Secondary Actors:	N/A
Trigger:	The admin selects a specific woodworker to view their registration details.		
Description:	This use case allows the admin to view detailed information about a specific woodworker's registration, including personal details, certifications, and other relevant data.		
Preconditions:	The admin is logged into the system. The admin has permission to view woodworker registration details.		
Post-conditions:	The system displays the full registration details of the selected woodworker.		
Normal Flow:	<ol style="list-style-type: none"> 1. The admin logs into the system. 2. The admin navigates to the "Đơn đăng ký xưởng" section. 3. The system displays the list of registered woodworkers. 		

	<p>4. The admin selects a woodworker from the list.</p> <p>5. The system displays the full registration details of the selected woodworker.</p>
Alternative Flows:	No Registration Data Available: <ul style="list-style-type: none">• If no data is available for the selected woodworker, the system notifies the admin.
Exceptions:	Network Error: <ul style="list-style-type: none">• If the registration details cannot be retrieved, an error message is displayed.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-90
Other Information:	The admin can export the woodworker registration details.
Assumptions:	Woodworker registration data is accurate and up-to-date in the system.

UC-86: Review registration

Use Case ID and Name:	UC-86: Review registration		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Admin	Secondary Actors:	N/A
Trigger:	The admin reviews a woodworker registration.		
Description:	This use case allows the admin to review a woodworker's registration information and determine whether to approve or reject the registration.		
Preconditions:	The admin is logged into the system. The admin has permission to review and approve/reject woodworker registrations.		
Post-conditions:	The system updates the status of the woodworker's registration (approved, rejected).		
Normal Flow:	<ol style="list-style-type: none"> 1. The admin logs into the system. 2. The admin navigates to the "Đơn đăng ký xưởng" section. 3. The system displays the list of registered woodworkers. 4. The admin selects a woodworker from the list. 5. The system displays the registration details. 6. The admin approves or rejects the registration. 		
Alternative Flows:	Registration Rejected: <ul style="list-style-type: none">• If the registration is rejected, the system will notify the woodworker.		
Exceptions:	Network Error:		

	<ul style="list-style-type: none"> If the registration cannot be processed, an error message is displayed.
Priority:	High
Frequency of Use:	Frequent
Business Rules:	BR-91
Other Information:	The admin can add comments when rejecting a registration.
Assumptions:	Woodworker registration data is complete and accurate.

UC-87: View configuration list

Use Case ID and Name:	UC-87: View Configuration List		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Admin	Secondary Actors:	N/A
Trigger:	The admin requests to view the list of system configurations.		
Description:	This use case allows the admin to view the list of system configurations, including settings and parameters related to the platform.		
Preconditions:	The admin is logged into the system. The admin has permission to view system configurations.		
Post-conditions:	The system displays the list of configurations.		
Normal Flow:	<ol style="list-style-type: none"> The admin logs into the system. The admin navigates to the "Cấu hình nền tảng" section. The system displays the list of configurations. 		
Alternative Flows:	No Configurations Available: <ul style="list-style-type: none"> If no configurations are available, the system notifies the admin. 		
Exceptions:	Network Error: <ul style="list-style-type: none"> If configuration data cannot be retrieved, an error message is displayed. 		
Priority:	High		
Frequency of Use:	Frequent		
Business Rules:	BR-92		
Other Information:	The admin can filter or sort the configurations.		
Assumptions:	Configuration data is stored and maintained accurately.		

UC-88: Update configuration

Use Case ID and Name:	UC-88: Update Configuration		
Created By:	Phúc	Date Created:	24/04/2025
Primary Actor:	Admin	Secondary Actors:	N/A
Trigger:	The admin updates a system configuration.		
Description:	This use case allows the admin to update system configurations, including parameters that control system settings and behavior.		
Preconditions:	The admin is logged into the system. The admin has permission to modify configurations.		
Post-conditions:	The system updates the configuration and applies the changes.		
Normal Flow:	<ol style="list-style-type: none"> The admin logs into the system. The admin navigates to the "Cáu hình nền tảng" section. The system displays the list of configurations. The admin selects a configuration to update. The system displays the configuration details. The admin modifies the configuration as needed. The system saves and applies the changes. 		
Alternative Flows:	<p>Invalid Configuration Update:</p> <ul style="list-style-type: none"> If the configuration update is invalid, the system will display an error and prompt the admin to correct it. 		
Exceptions:	<p>Network Error:</p> <ul style="list-style-type: none"> If the configuration cannot be updated due to network issues, an error message is displayed. 		
Priority:	High		
Frequency of Use:	Moderate		
Business Rules:	BR-93		
Other Information:	The admin can reset the configuration to its default state if needed.		
Assumptions:	Configuration data is accurate and up-to-date.		

3. Functional Requirements

3.1 System Functional Overview

3.1.1 Customer Web Application / Mobile Application Screen Flow

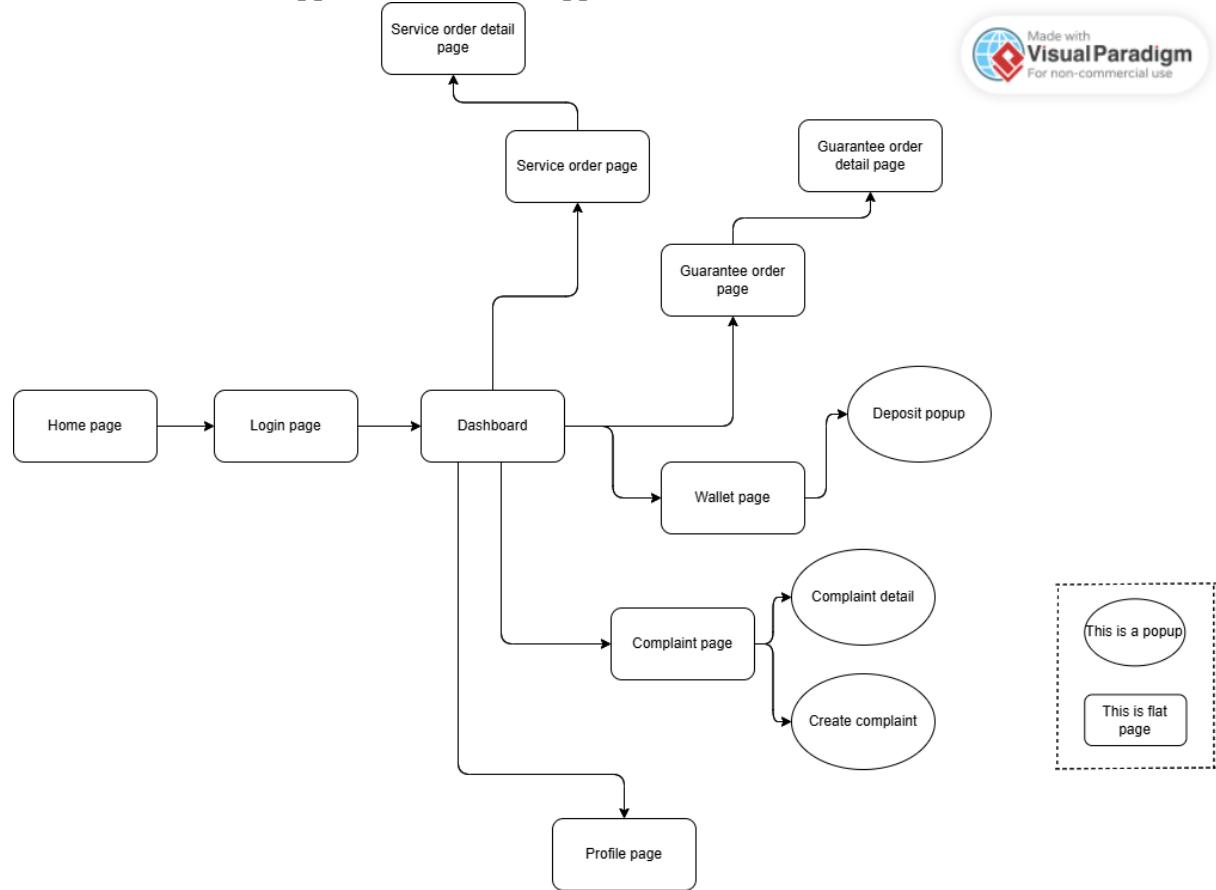


Figure 5 - Customer Web Application / Mobile Application Screen Flow

3.1.2 Woodworker Web Application / Mobile Application Screen Flow

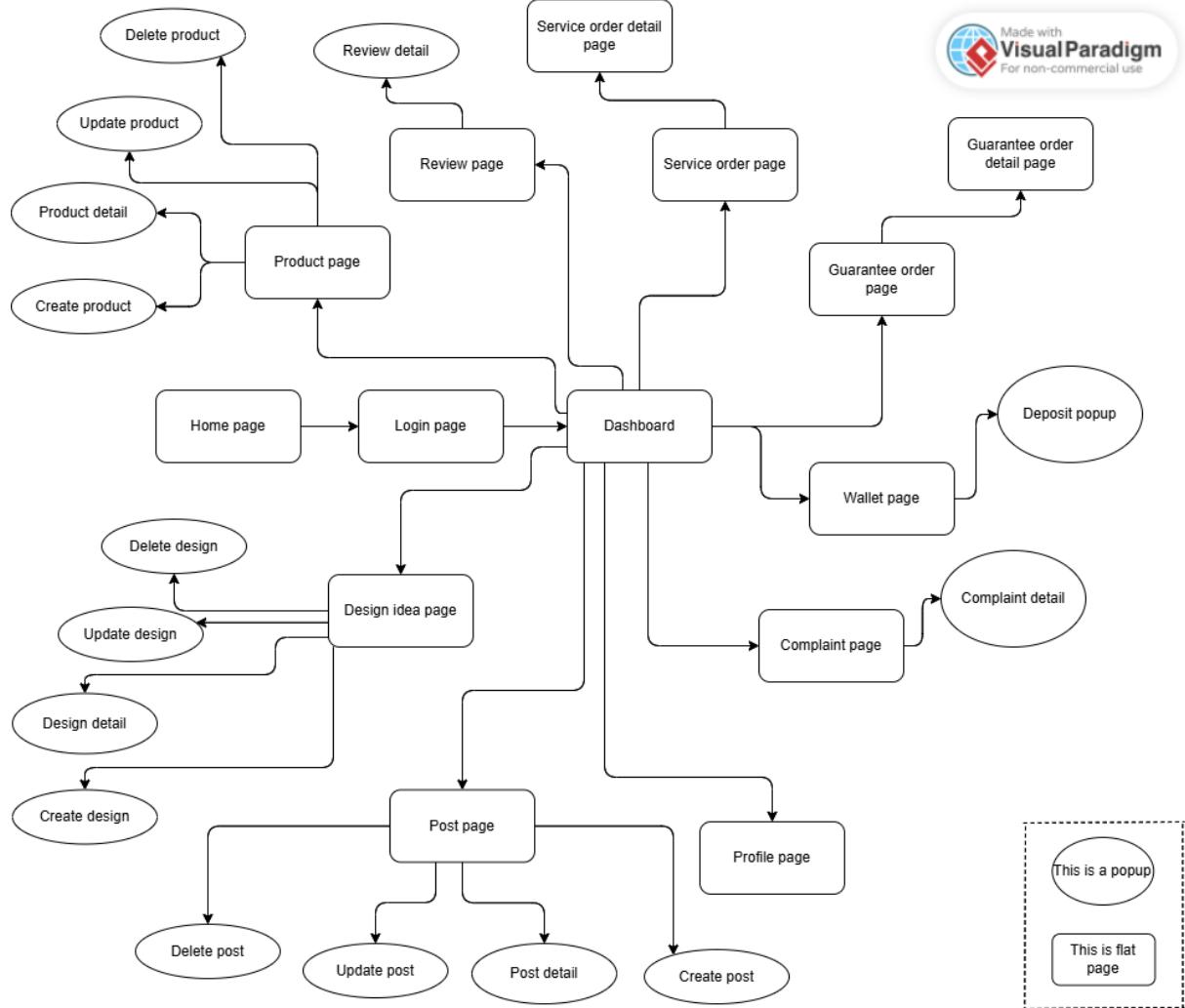


Figure 6 - Woodworker Web Application / Mobile Application Screen Flow

3.1.3 Staff Web Application Screen Flow

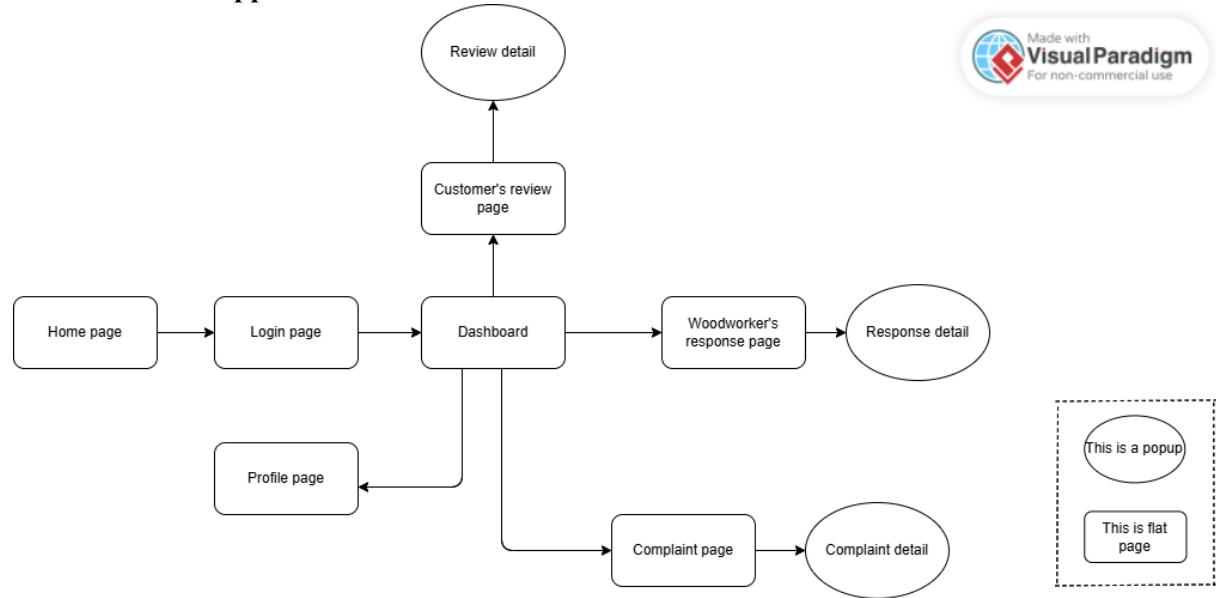


Figure 7 - Staff Web Application Screen Flow

3.1.4 Moderator Web Application Screen Flow

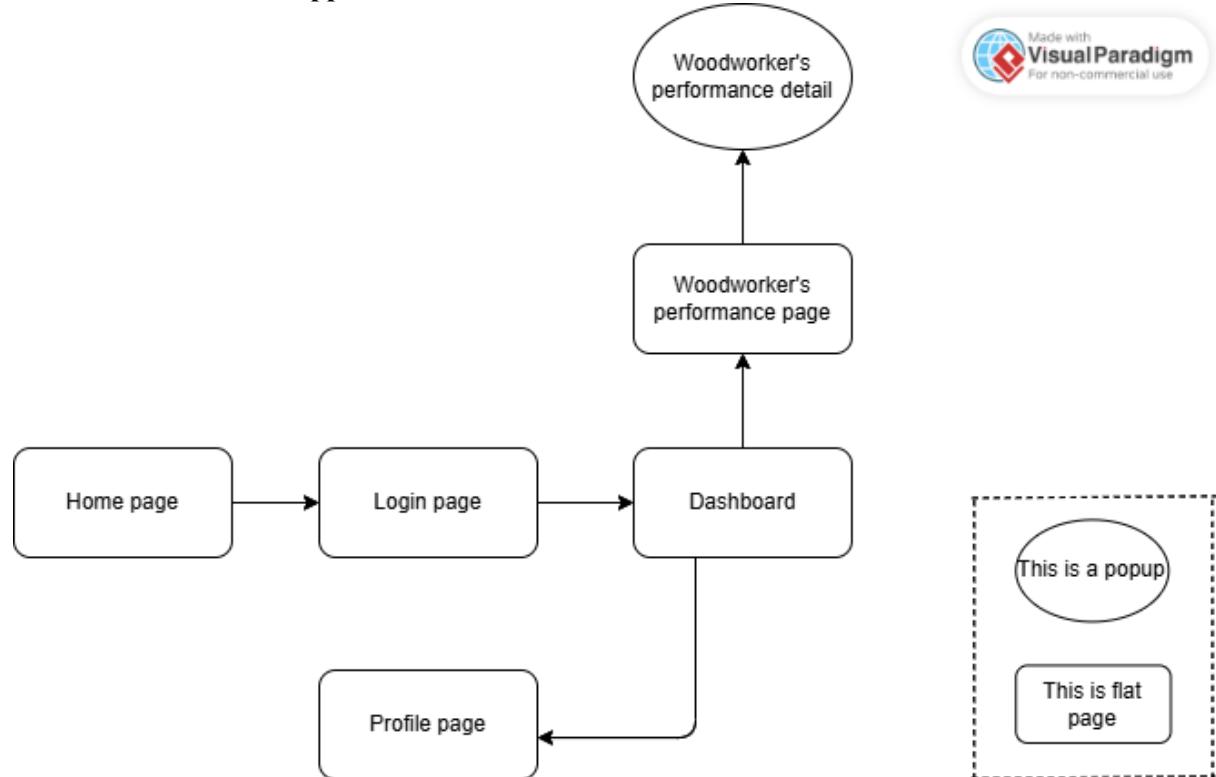


Figure 8 - Moderator Web Application Screen Flow

3.1.5 Admin Web Application Screen Flow

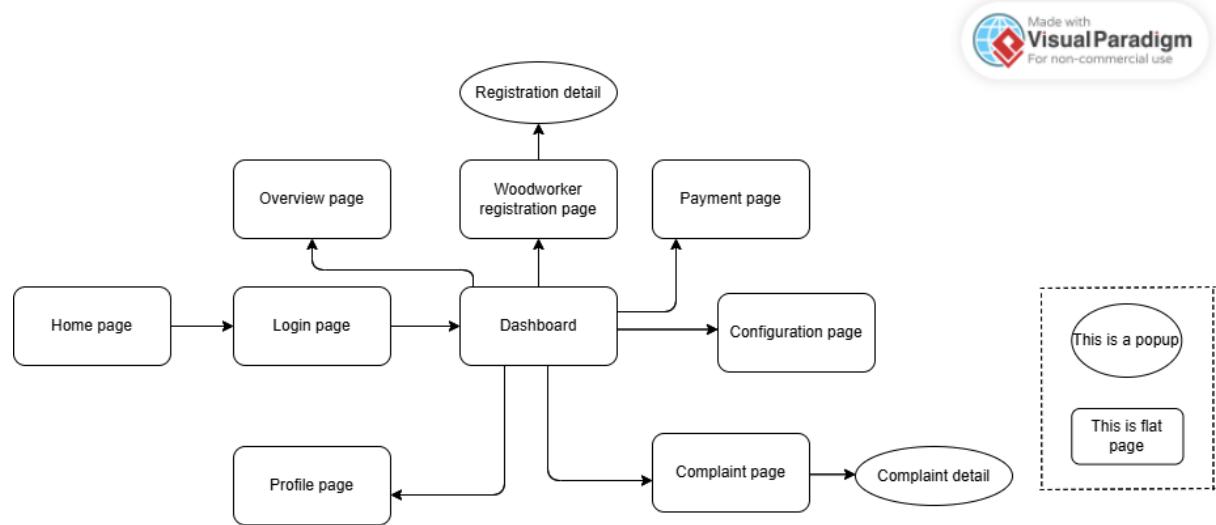


Figure 9 - Admin Web Application Screen Flow

3.2 Entity Relationship Diagram

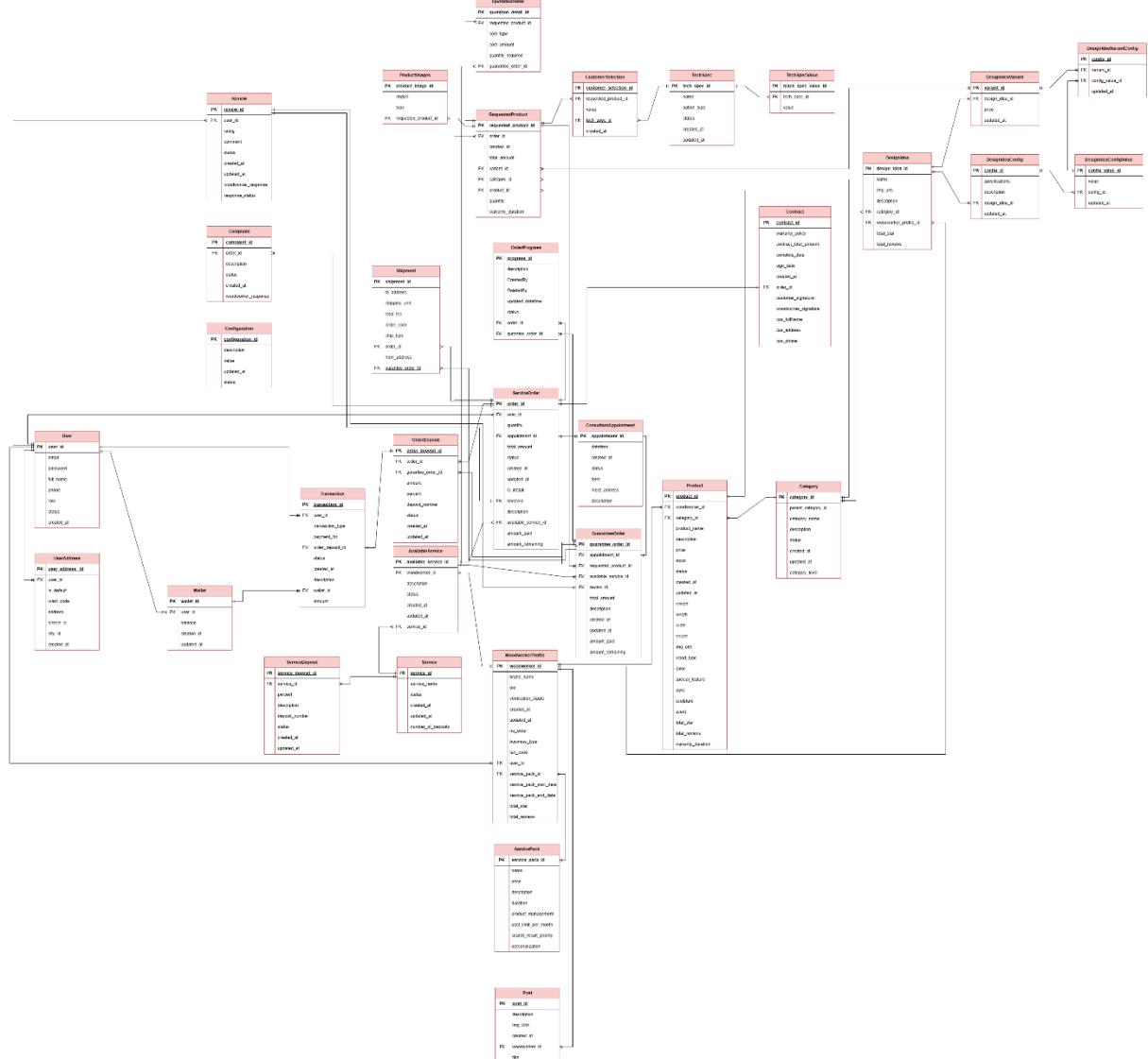


Figure 10 - Entity Relationship Diagram

Entity name	Attributes	Description
Category	category_id	<ul style="list-style-type: none"> A unique identifier for each category. This serves as the primary key (PK) for the table, ensuring that each category record is distinct.
	parent_category_id	<ul style="list-style-type: none"> The unique identifier of the parent category under which the current category falls. This foreign key links a category to its parent, enabling hierarchical categorization.
	category_name	<ul style="list-style-type: none"> The name of the category. This attribute is used to represent the title or label of the category, helping to identify and describe

		the type of items or services it contains.
	description	<ul style="list-style-type: none"> A detailed textual description of the category. This attribute provides additional information about the category, including its purpose, scope, and the types of items or services it includes.
	status	<ul style="list-style-type: none"> The current status of the category (e.g., active, inactive, archived). This helps track whether the category is currently available for use in the system.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the category was created. This helps track when the category was added to the system.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the category was updated. This is used to track any modifications made to the category after its creation.
	category_level	<ul style="list-style-type: none"> The level or depth of the category within the hierarchical structure (e.g., 1 for top-level categories, 2 for subcategories). This helps organize categories and understand their position in the category hierarchy.
Configuration	configuration_id	<ul style="list-style-type: none"> A unique identifier for each configuration record. This serves as the primary key (PK) for the table, ensuring that each configuration entry is distinct.
	description	<ul style="list-style-type: none"> A detailed textual description of the configuration. This attribute provides information about the purpose or context of the configuration and what it controls or defines.
	value	<ul style="list-style-type: none"> The actual value associated with the configuration. This can be a setting, parameter, or option that the configuration defines, and it can vary depending on the type of configuration (e.g., numeric value, string, or boolean).

	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the configuration was updated. This helps track when changes were made to the configuration.
	status	<ul style="list-style-type: none"> The current status of the configuration (e.g., active, inactive, disabled). This attribute indicates whether the configuration is currently in use or has been deactivated.
ConsultantAppointment	appointment_id	<ul style="list-style-type: none"> A unique identifier for each consultant appointment. This serves as the primary key (PK) for the table, ensuring that each appointment record is distinct.
	datetime	<ul style="list-style-type: none"> The date and time the appointment is scheduled to take place. This attribute helps determine the time of the meeting with the consultant.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the appointment was created. This helps track when the appointment was booked in the system.
	status	<ul style="list-style-type: none"> The current status of the appointment (e.g., scheduled, completed, canceled, pending). This attribute helps track the progress or state of the appointment.
	form	<ul style="list-style-type: none"> The type of form or medium used for the appointment (e.g., in-person, virtual). This helps indicate how the appointment is being conducted.
	meet_address	<ul style="list-style-type: none"> The physical address or location where the appointment will take place if it's an in-person meeting. This attribute is used to specify where the consultant and client will meet.

	description	<ul style="list-style-type: none"> • A detailed description or additional information about the appointment. • This attribute can be used to provide context, the purpose of the meeting, or any relevant notes regarding the appointment.
Service	service_id	<ul style="list-style-type: none"> • A unique identifier for each service. • This serves as the primary key (PK) for the table and ensures that each service record is distinct.
	service_name	<ul style="list-style-type: none"> • The name of the service offered by the woodworker. Examples of services could include "Customization," "Personalization," or "Guarantee Service Booking." • This attribute is used to identify and distinguish between different types of services.
	status	<ul style="list-style-type: none"> • The current status of the service (e.g., active, inactive, suspended). • This helps track whether the service is currently available or if it has been deactivated or put on hold.
	created_at	<ul style="list-style-type: none"> • The timestamp indicating when the service was created. • This helps track when the service was added to the system.
	updated_at	<ul style="list-style-type: none"> • The timestamp indicating the last time the service record was updated. • This is used to track changes made to the service after its creation.
	number_of_deposits	<ul style="list-style-type: none"> • The total number of deposits or pre-payments made for the service. • This attribute is useful in tracking the financial commitments related to the service booking, such as deposits for customization or guarantee services.

ServiceDeposit	service_deposit_id	<ul style="list-style-type: none"> A unique identifier for each service deposit record. This serves as the primary key (PK) for the table, ensuring that each deposit entry is distinct.
	service_id	<ul style="list-style-type: none"> The unique identifier for the service associated with the deposit. This foreign key links the deposit to a specific service in the system (e.g., Customization, Personalization, or Guarantee Service Booking).
	percent	<ul style="list-style-type: none"> The percentage of the total service cost that is required as a deposit. This attribute specifies how much of the service price needs to be paid upfront as a deposit.
	description	<ul style="list-style-type: none"> A textual description providing details about the deposit. This could include the terms of the deposit, such as conditions for refunds or payment instructions.
	deposit_number	<ul style="list-style-type: none"> The total number of deposits or payments that need to be made for the service. This could be used if there are multiple installment payments or stages for the deposit.
	status	<ul style="list-style-type: none"> The current status of the deposit (e.g., pending, paid, canceled). This helps track whether the deposit has been made, is awaiting payment, or has been canceled.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the deposit record was created. This helps track when the deposit requirement was added to the system.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the deposit record was updated. This is used to track changes made to the deposit details after its initial creation.

ServicePack	service_pack_id	<ul style="list-style-type: none"> • A unique identifier for each service pack. • This serves as the primary key (PK) for the table, ensuring that each service pack record is distinct.
	name	<ul style="list-style-type: none"> • The name of the service pack. • This attribute is used to identify the service pack and distinguish it from other service offerings.
	price	<ul style="list-style-type: none"> • The price of the service pack. • This attribute represents the cost of subscribing to or purchasing the service pack.
	description	<ul style="list-style-type: none"> • A detailed description of the service pack. • This attribute provides additional information about what the service pack includes, its features, and benefits.
	duration	<ul style="list-style-type: none"> • The length of time for which the service pack is valid. • This attribute specifies the period during which the service pack provides access to its services (e.g., 1 month, 1 year).
	product_management	<ul style="list-style-type: none"> • A boolean or categorical attribute indicating whether the service pack includes product management features or services. • It helps define whether product management is part of the package.
	post_limit_per_month	<ul style="list-style-type: none"> • The maximum number of posts or services that can be utilized per month under the service pack. • This attribute helps define usage limitations associated with the service pack.
	search_result_priority	<ul style="list-style-type: none"> • The priority or ranking of the service pack in search results. • This attribute determines the visibility or prominence of the service pack when users search for services.
	personalization	<ul style="list-style-type: none"> • A boolean or categorical attribute indicating whether the service pack includes personalization features, such

		as customized services or tailored offerings for users.
TechSpec	tech_spec_id	<ul style="list-style-type: none"> A unique identifier for each technical specification record. This serves as the primary key (PK) for the table, ensuring that each tech spec entry is distinct.
	name	<ul style="list-style-type: none"> The name of the technical specification. This attribute represents the title or label of the tech spec, such as a specific feature or characteristic of a product or service.
	option_type	<ul style="list-style-type: none"> The type of option available for the technical specification. This could represent a set of predefined options (e.g., size, color, capacity) or a custom input field, defining how the tech spec should be selected or input.
	status	<ul style="list-style-type: none"> The current status of the technical specification (e.g., active, inactive, deprecated). This helps track whether the tech spec is currently available or has been disabled.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the technical specification was created. This helps track when the tech spec was added to the system.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the technical specification was updated. This is used to track changes made to the tech spec after its initial creation.
TechSpecValue	tech_spec_value_id	<ul style="list-style-type: none"> A unique identifier for each technical specification value record. This serves as the primary key (PK) for the table, ensuring that each entry is distinct.
	tech_spec_id	<ul style="list-style-type: none"> The unique identifier of the technical specification that this value belongs to. This foreign key links the value to a particular technical specification, helping organize and relate the value to its spec.

	value	<ul style="list-style-type: none"> The actual value for the technical specification. This could be a numerical value, text, or any other data type that corresponds to the specific tech spec (e.g., "256GB" for storage, "Red" for color).
User	user_id	<ul style="list-style-type: none"> A unique identifier for the user, typically used as the primary key.
	email	<ul style="list-style-type: none"> The user's email address. It is typically used as the primary communication channel and often serves as the unique identifier for the user in the system.
	password	<ul style="list-style-type: none"> The user's password. This attribute stores the hashed password that is used to authenticate the user during login.
	full_name	<ul style="list-style-type: none"> A fullname chosen by the user to identify themselves within the system. It is used as a display name.
	phone	<ul style="list-style-type: none"> The user's phone number. This attribute is used for communication purposes and may be used in account recovery or two-factor authentication.
	role	<ul style="list-style-type: none"> The role assigned to the user within the system. It defines the user's permissions and access levels (e.g., Admin, User, Woodworker).
	status	<ul style="list-style-type: none"> The current status of the user account (e.g., active, suspended, deactivated). This attribute helps track the user's account state within the system.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the user account was created. This attribute helps track when the user registered or joined the system.
Review	review_id	<ul style="list-style-type: none"> A unique identifier for each review.

		<ul style="list-style-type: none"> It is the primary key (PK) and is used to reference individual reviews in the system.
	user_id	<ul style="list-style-type: none"> The unique identifier of the user who created the review. This foreign key links the review to the user who submitted it.
	rating	<ul style="list-style-type: none"> A numerical value given by the user to rate the reviewed item, typically on a scale (e.g., 1 to 5). This helps assess the quality or satisfaction level of the reviewed item or service.
	comment	<ul style="list-style-type: none"> The textual feedback or opinion provided by the user about the reviewed item. This allows the user to elaborate on their rating and provide more detailed insights.
	status	<ul style="list-style-type: none"> The current status of the review (e.g., pending, approved, rejected). This attribute is used to track whether the review is under review, published, or removed.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the review was created. This helps track when the review was submitted in relation to other system events.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the review was updated. It is used to track changes made to the review after its initial submission.
	woodworker_response	<ul style="list-style-type: none"> The response or feedback provided by the woodworker (or service provider) to address the user's review or comments. This is typically used in services where the provider replies to user feedback.
	response_status	<ul style="list-style-type: none"> The status of the woodworker's response (e.g., pending, addressed, resolved).

		<ul style="list-style-type: none"> • This helps track whether the woodworker's reply is still awaiting approval or action.
UserAddress	user_address_id	<ul style="list-style-type: none"> • A unique identifier for each user address record. • It serves as the primary key (PK) for the table, ensuring that each address entry is distinct.
	user_id	<ul style="list-style-type: none"> • The unique identifier of the user who owns the address. • This foreign key links the address to the specific user in the system.
	is_default	<ul style="list-style-type: none"> • A boolean value indicating whether the address is the user's default address. • This attribute helps determine which address should be used for default shipping or billing purposes.
	ward_code	<ul style="list-style-type: none"> • The code representing the ward or sub-district in which the user's address is located. • This is typically used to specify smaller geographic regions within a city or district.
	address	<ul style="list-style-type: none"> • The full address of the user, including street name, house number, or other specific details needed to locate the address.
	district_id	<ul style="list-style-type: none"> • The unique identifier of the district or city subdivision in which the address is located. • This foreign key is used to link the address to a specific district within the city.
	city_id	<ul style="list-style-type: none"> • The unique identifier of the city where the user's address is located. • This foreign key links the address to a specific city.
	created_at	<ul style="list-style-type: none"> • The timestamp indicating when the user address was created.

		<ul style="list-style-type: none"> • This helps track when the address was added to the system.
Wallet	wallet_id	<ul style="list-style-type: none"> • A unique identifier for each wallet record. • It serves as the primary key (PK) for the table, ensuring that each wallet entry is distinct.
	user_id	<ul style="list-style-type: none"> • The unique identifier of the user who owns the wallet. • This foreign key links the wallet to the specific user in the system.
	balance	<ul style="list-style-type: none"> • The current balance in the wallet. • It represents the amount of funds available for transactions or withdrawals in the user's wallet.
	created_at	<ul style="list-style-type: none"> • The timestamp indicating when the wallet was created. • This helps track when the wallet was initialized or opened for the user.
	updated_at	<ul style="list-style-type: none"> • The timestamp indicating the last time the wallet was updated. • This is used to track any changes to the wallet's balance or other information after its creation.
WoodworkerProfile	woodworker_id	<ul style="list-style-type: none"> • A unique identifier for each woodworker profile. • It serves as the primary key (PK) for the table, ensuring that each woodworker's profile is distinct.
	brand_name	<ul style="list-style-type: none"> • The name of the brand or business associated with the woodworker. • This is used for branding and identification purposes in the system.
	bio	<ul style="list-style-type: none"> • A brief description or biography of the woodworker. • This attribute provides additional information about the woodworker, their experience, and their specialization.

	verification_status	<ul style="list-style-type: none"> The status indicating whether the woodworker's profile has been verified. It can have values such as 'verified' or 'unverified', and it is used to determine if the woodworker has been authenticated by the platform.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the woodworker's profile was created. This helps track when the woodworker joined the platform.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the woodworker's profile was updated. This is used to track changes made to the profile after its initial creation.
	no_order	<ul style="list-style-type: none"> The number of orders or projects that the woodworker has completed. This helps track the woodworker's activity and experience on the platform.
	business_type	<ul style="list-style-type: none"> The type of business or services provided by the woodworker (e.g., individual, company, etc.). This attribute defines the woodworker's business model.
	tax_code	<ul style="list-style-type: none"> The tax identification number or code of the woodworker's business. This is used for legal and tax-related purposes.
	user_id	<ul style="list-style-type: none"> The unique identifier of the user who owns the woodworker profile. This foreign key links the woodworker profile to the corresponding user in the system.
	service_pack_id	<ul style="list-style-type: none"> The unique identifier for the service pack associated with the woodworker. This links the woodworker to a specific service package they

		have chosen or are subscribed to.
	service_pack_start_date	<ul style="list-style-type: none"> The start date of the woodworker's service pack subscription. This helps track the period during which the service pack is active.
	service_pack_end_date	<ul style="list-style-type: none"> The end date of the woodworker's service pack subscription. This is used to determine when the service pack expires and needs renewal.
	total_star	<ul style="list-style-type: none"> The total number of stars or rating points that the woodworker has received from users based on their reviews. This is used to evaluate the woodworker's overall performance.
	total_review	<ul style="list-style-type: none"> The total number of reviews that the woodworker has received from users. This helps measure the woodworker's reputation and user feedback.
AvailableService	available_service_id	<ul style="list-style-type: none"> A unique identifier for each available service record. It serves as the primary key (PK) for the table, ensuring that each service entry is distinct.
	woodworker_id	<ul style="list-style-type: none"> The unique identifier of the woodworker who offers the service. This foreign key links the service to the specific woodworker providing it.
	description	<ul style="list-style-type: none"> A detailed description of the available service. This attribute provides information about what the service entails and any specific details or features of the offering.
	status	<ul style="list-style-type: none"> The current status of the available service (e.g., active, inactive, suspended).

		<ul style="list-style-type: none"> This helps track whether the service is currently available for users to choose.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the available service was created. This helps track when the service was added to the system.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the available service record was updated. This is used to track any modifications made to the service after its creation.
	service_id	<ul style="list-style-type: none"> The unique identifier for the service being offered. This foreign key links the available service to a specific predefined service in the system, such as a type of woodwork service.
DesignIdea	design_idea_id	<ul style="list-style-type: none"> A unique identifier for each design idea. It serves as the primary key (PK) for the table, ensuring that each design idea record is distinct.
	name	<ul style="list-style-type: none"> The name of the design idea. This attribute is used to represent the title or label of the design idea in the system.
	img_urls	<ul style="list-style-type: none"> A list or array of URLs pointing to images associated with the design idea. These images provide a visual representation of the design idea.
	description	<ul style="list-style-type: none"> A detailed textual description of the design idea. This attribute elaborates on the design, its features, and any relevant details for users to understand the concept.
	category_id	<ul style="list-style-type: none"> The unique identifier of the category under which the design idea falls. This foreign key links the design idea to a specific category, helping organize

		design ideas into relevant groups.
	woodworker_profile_id	<ul style="list-style-type: none"> The unique identifier of the woodworker's profile associated with the design idea. This foreign key links the design idea to the specific woodworker who created or submitted the design.
	total_star	<ul style="list-style-type: none"> The total number of stars or rating points that the design idea has received from users. This attribute helps measure the popularity and quality of the design idea based on user feedback.
	total_reviews	<ul style="list-style-type: none"> The total number of reviews or ratings that the design idea has received. This attribute helps quantify the feedback and overall user interaction with the design idea.
DesignIdeaConfig	config_id	<ul style="list-style-type: none"> A unique identifier for each design idea configuration. This serves as the primary key (PK) for the table and ensures that each configuration entry is distinct.
	specifications	<ul style="list-style-type: none"> Detailed specifications for the design idea configuration. This attribute describes the technical or material details associated with the design, such as dimensions, materials, or features.
	description	<ul style="list-style-type: none"> A textual description of the design idea configuration. It provides additional details or context about the configuration and its intended purpose or usage.
	design_idea_id	<ul style="list-style-type: none"> The unique identifier of the design idea associated with the configuration. This foreign key links the configuration to the design idea it belongs to.

	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the design idea configuration was updated. This helps track when the configuration was last modified.
DesignIdeaConfigValue	config_value_id	<ul style="list-style-type: none"> A unique identifier for each design idea configuration value record. This serves as the primary key (PK) for the table, ensuring that each entry is distinct.
	value	<ul style="list-style-type: none"> The actual value associated with the design idea configuration. This could be a numeric, text, or other type of value that defines a specific characteristic or setting for the design idea configuration.
	config_id	<ul style="list-style-type: none"> The unique identifier for the design idea configuration that this value belongs to. This foreign key links the configuration value to a specific design idea configuration, allowing for the organization of multiple values under one configuration.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the design idea configuration value was updated. This helps track when changes were made to the configuration value after its creation.
DesignIdeaVariant	variant_id	<ul style="list-style-type: none"> A unique identifier for each design idea variant. This serves as the primary key (PK) for the table, ensuring that each variant entry is distinct.
	design_idea_id	<ul style="list-style-type: none"> The unique identifier of the design idea associated with the variant. This foreign key links the variant to a specific design idea in the system.
	price	<ul style="list-style-type: none"> The price associated with the design idea variant.

		<ul style="list-style-type: none"> This attribute specifies the cost of the particular variant of the design idea, which may differ depending on the specifications or customization options.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the design idea variant was updated. This helps track when the variant's details, such as price or features, were modified.
DesignIdeaVariantConfig	config_id	<ul style="list-style-type: none"> A unique identifier for each design idea variant configuration record. This serves as the primary key (PK) for the table, ensuring that each configuration entry is distinct.
	variant_id	<ul style="list-style-type: none"> The unique identifier of the design idea variant associated with this configuration. This foreign key links the configuration to a specific variant of a design idea, allowing for variant-specific settings.
	config_value_id	<ul style="list-style-type: none"> The unique identifier of the configuration value associated with the design idea variant. This foreign key links the configuration to a specific configuration value, defining specific properties or characteristics for the variant.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the design idea variant configuration was updated. This helps track when any changes were made to the configuration or its associated values.
Post	post_id	<ul style="list-style-type: none"> A unique identifier for each post. This serves as the primary key (PK) for the tab each post record is distinct.
	description	<ul style="list-style-type: none"> A textual description of the post.

		<ul style="list-style-type: none"> This attribute provides detailed information about the content of the post, which could include announcements, updates, or other relevant information related to the woodworker or service.
	imgUrls	<ul style="list-style-type: none"> A list or array of URLs pointing to images associated with the post. These images provide visual context or additional details related to the post's content.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the post was created. This helps track when the post was published or added to the system.
	woodworker_id	<ul style="list-style-type: none"> The unique identifier of the woodworker who created the post. This foreign key links the post to a specific woodworker, indicating the origin of the post.
	title	<ul style="list-style-type: none"> The title or heading of the post. This attribute serves as a short summary or headline for the content of the post, helping users quickly understand its topic or focus.
Product	product_id	<ul style="list-style-type: none"> A unique identifier for each product. This serves as the primary key (PK) for the table, ensuring that each product record is distinct.
	woodworker_id	<ul style="list-style-type: none"> The unique identifier of the woodworker who created or is selling the product. This foreign key links the product to a specific woodworker in the system
	category_id	<ul style="list-style-type: none"> The unique identifier of the category under which the product falls. This foreign key links the product to a specific category,

		helping organize the products into relevant groups.
	product_name	<ul style="list-style-type: none"> The name or title of the product. This attribute is used to identify the product and differentiate it from other products in the system.
	description	<ul style="list-style-type: none"> A detailed textual description of the product. This provides information about the product's features, uses, and other relevant details.
	price	<ul style="list-style-type: none"> The price of the product. This attribute specifies how much the product costs.
	stock	<ul style="list-style-type: none"> The number of units available for the product. This attribute indicates the current stock level and helps manage inventory.
	status	<ul style="list-style-type: none"> The current status of the product (e.g., active, inactive, out of stock). This helps track whether the product is available for sale or not.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the product was created. This helps track when the product was added to the system.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the product information was updated. This is used to track any changes made to the product after its initial creation.
	weight	<ul style="list-style-type: none"> The weight of the product. This attribute is used to specify how heavy the product is, which may be relevant for shipping or handling.
	length	<ul style="list-style-type: none"> The length of the product. This attribute helps define the product's dimensions.
	width	<ul style="list-style-type: none"> The width of the product. This attribute helps define the product's dimensions.
	height	<ul style="list-style-type: none"> The height of the product.

		<ul style="list-style-type: none"> This attribute helps define the product's dimensions.
	img_urls	<ul style="list-style-type: none"> A list or array of URLs pointing to images associated with the product. These images provide a visual representation of the product.
	wood_type	<ul style="list-style-type: none"> The type of wood used to make the product. This attribute defines the material composition of the product.
	color	<ul style="list-style-type: none"> The color of the product. This attribute specifies the product's appearance.
	special_feature	<ul style="list-style-type: none"> A list of special features or unique qualities of the product. This could include specific functional or aesthetic attributes that distinguish the product.
	style	<ul style="list-style-type: none"> The style or design category of the product. This could refer to the overall aesthetic or design theme of the product (e.g., modern, vintage).
	sculpture	<ul style="list-style-type: none"> Indicates whether the product includes sculptural elements or is a sculpture itself. This attribute may apply to artistic or decorative products.
	scent	<ul style="list-style-type: none"> The scent of the product, if applicable. This attribute is relevant for products like candles or wood-related products that have a specific smell.
	total_star	<ul style="list-style-type: none"> The total number of stars or rating points that the product has received based on user reviews. This helps assess the overall quality or popularity of the product.
	total_reviews	<ul style="list-style-type: none"> The total number of reviews that the product has received.

		<ul style="list-style-type: none"> This attribute helps measure the amount of user feedback or interaction with the product.
	warranty_duration	<ul style="list-style-type: none"> The length of time for which the product is covered under warranty. This attribute specifies the duration (e.g., 1 year, 5 years) that the manufacturer guarantees the product's quality.
ServiceOrder	order_id	<ul style="list-style-type: none"> A unique identifier for each service order. This serves as the primary key (PK) for the table, ensuring that each order record is distinct.
	user_id	<ul style="list-style-type: none"> The unique identifier of the user who placed the service order. This foreign key links the order to a specific user in the system.
	quantity	<ul style="list-style-type: none"> The quantity of the service or product ordered. This attribute specifies how many units of the service are requested.
	appointment_id	<ul style="list-style-type: none"> The unique identifier of the appointment associated with the service order. This foreign key links the order to a specific appointment, indicating when the service will take place.
	total_amount	<ul style="list-style-type: none"> The total amount for the service order. This attribute represents the full cost of the service or product, including any additional fees or charges.
	status	<ul style="list-style-type: none"> The current status of the service order (e.g., pending, completed, canceled). This helps track the progress of the order and its fulfillment.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the service order was created.

		<ul style="list-style-type: none"> • This helps track when the order was placed in the system.
	updated_at	<ul style="list-style-type: none"> • The timestamp indicating the last time the service order was updated. • This is used to track changes made to the order after its creation.
	is_install	<ul style="list-style-type: none"> • A boolean value indicating whether the service order includes installation services. • This attribute helps determine if additional installation services are required for the order.
	review_id	<ul style="list-style-type: none"> • The unique identifier of the review associated with the service order. • This foreign key links the order to a specific review, allowing the user to provide feedback after the service is completed.
	description	<ul style="list-style-type: none"> • A detailed textual description of the service order. • This attribute can include any additional notes or specifications provided by the user or service provider.
	available_service_id	<ul style="list-style-type: none"> • The unique identifier of the available service associated with the order. • This foreign key links the order to a specific service from the list of available services.
	amount_paid	<ul style="list-style-type: none"> • The amount that has already been paid for the service order. • This attribute tracks the payment status of the order.
	amount_remaining	<ul style="list-style-type: none"> • The remaining amount that still needs to be paid for the service order. • This attribute helps track the outstanding balance that needs to be settled.
Contract	contract_id	<ul style="list-style-type: none"> • A unique identifier for each contract.

		<ul style="list-style-type: none"> This serves as the primary key (PK) for the table, ensuring that each contract record is distinct.
	warranty_policy	<ul style="list-style-type: none"> The warranty policy associated with the contract. This attribute outlines the terms and conditions of the warranty, such as the duration and coverage details.
	contract_total_amount	<ul style="list-style-type: none"> The total amount specified in the contract. This represents the overall value of the contract, including services, products, or other costs associated with the agreement.
	complete_date	<ul style="list-style-type: none"> The date when the contract's services or obligations were completed. This marks the end of the contract's fulfillment phase.
	sign_date	<ul style="list-style-type: none"> The date when the contract was signed by both parties (customer and woodworker). This signifies the formal agreement of the contract terms.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the contract was created in the system. This helps track when the contract was first initiated.
	order_id	<ul style="list-style-type: none"> The unique identifier of the order associated with the contract. This foreign key links the contract to a specific service order in the system.
	customer_signature	<ul style="list-style-type: none"> The digital or physical signature of the customer. This attribute records the customer's agreement and consent to the terms outlined in the contract.
	woodworker_signature	<ul style="list-style-type: none"> The digital or physical signature of the woodworker. This attribute records the woodworker's agreement and consent to the terms of the contract.

	cus_fullName	<ul style="list-style-type: none"> The full name of the customer. This attribute is used to identify the customer involved in the contract.
	cus_address	<ul style="list-style-type: none"> The address of the customer. This attribute records the customer's residential or service address as part of the contract.
	cus_phone	<ul style="list-style-type: none"> The phone number of the customer. This attribute allows contact with the customer for communication related to the contract.
RequestedProduct	requested_product_id	<ul style="list-style-type: none"> A unique identifier for each requested product. This serves as the primary key (PK) for the table, ensuring that each product request record is distinct.
	order_id	<ul style="list-style-type: none"> The unique identifier of the order associated with the requested product. This foreign key links the requested product to a specific service or purchase order in the system.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the requested product was added to the system. This helps track when the product request was made.
	total_amount	<ul style="list-style-type: none"> The total amount for the requested product. This attribute represents the full cost for the specific product requested, including any additional charges or fees.
	variant_id	<ul style="list-style-type: none"> The unique identifier for the variant of the product requested. This foreign key links the requested product to a specific variant, such as a particular style, color, or size.
	category_id	<ul style="list-style-type: none"> The unique identifier of the category under which the requested product falls.

		<ul style="list-style-type: none"> This foreign key links the product to a specific category, helping organize the product within the system.
	product_id	<ul style="list-style-type: none"> The unique identifier of the product being requested. This foreign key links the requested product to a specific product in the system.
	quantity	<ul style="list-style-type: none"> The number of units of the requested product. This attribute indicates how many units of the product are being requested in the order.
	warranty_duration	<ul style="list-style-type: none"> The length of time for which the requested product is covered under warranty. This attribute specifies the duration (e.g., 1 year, 5 years) that the manufacturer guarantees the product's quality.
CustomerSelection	customer_selection_id	<ul style="list-style-type: none"> A unique identifier for each customer selection record. This serves as the primary key (PK) for the table, ensuring that each selection entry is distinct.
	requested_product_id	<ul style="list-style-type: none"> The unique identifier of the requested product associated with the customer selection. This foreign key links the selection to a specific requested product in the system.
	value	<ul style="list-style-type: none"> The value selected by the customer for a specific attribute or option related to the product. This could be a choice like color, size, or any other configurable feature of the product.
	tech_spec_id	<ul style="list-style-type: none"> The unique identifier of the technical specification associated with the customer selection. This foreign key links the selection to a specific technical specification (e.g.,

		material, capacity, or any other technical detail).
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the customer selection was made. This helps track when the customer made their choice for the product.
GuaranteeOrder	guarantee_order_id	<ul style="list-style-type: none"> A unique identifier for each guarantee order. This serves as the primary key (PK) for the table, ensuring that each guarantee order record is distinct.
	appointment_id	<ul style="list-style-type: none"> The unique identifier of the appointment associated with the guarantee order. This foreign key links the guarantee order to a specific appointment for service or resolution.
	requested_product_id	<ul style="list-style-type: none"> The unique identifier of the requested product associated with the guarantee order. This foreign key links the order to the specific product that is under guarantee.
	available_service_id	<ul style="list-style-type: none"> The unique identifier of the available service associated with the guarantee order. This foreign key links the guarantee order to a specific service, such as repair or replacement, offered to the customer.
	review_id	<ul style="list-style-type: none"> The unique identifier of the review associated with the guarantee order. This foreign key links the guarantee order to a customer's review of the service provided.
	total_amount	<ul style="list-style-type: none"> The total amount for the guarantee order. This attribute represents the full cost of the guarantee service, including any fees for repairs, replacements, or other services.
	description	<ul style="list-style-type: none"> A detailed textual description of the guarantee order.

		<ul style="list-style-type: none"> This attribute can include specific details about the issue being addressed, the service being provided, or any other relevant information.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the guarantee order was created. This helps track when the order was initiated in the system.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the guarantee order was updated. This is used to track any changes made to the order after its creation.
	amount_paid	<ul style="list-style-type: none"> The amount that has already been paid for the guarantee order. This attribute tracks the payment status for the guarantee service.
	amount_remaining	<ul style="list-style-type: none"> The remaining amount that still needs to be paid for the guarantee order. This helps track the outstanding balance that needs to be settled.
OrderDeposit	order_deposit_id	<ul style="list-style-type: none"> A unique identifier for each order deposit record. This serves as the primary key (PK) for the table, ensuring that each deposit entry is distinct.
	order_id	<ul style="list-style-type: none"> The unique identifier of the order associated with the deposit. This foreign key links the deposit to a specific order in the system.
	guarantee_order_id	<ul style="list-style-type: none"> The unique identifier of the guarantee order associated with the deposit. This foreign key links the deposit to a specific guarantee order in the system.

	amount	<ul style="list-style-type: none"> The total amount of the deposit paid for the order or guarantee order. This represents the upfront payment made by the customer.
	percent	<ul style="list-style-type: none"> The percentage of the total order or guarantee order amount that the deposit represents. This attribute helps define how much of the order's total value is covered by the deposit.
	deposit_number	<ul style="list-style-type: none"> The total number of deposits or installment payments made or required for the order. This attribute tracks how many payments have been made or are planned.
	status	<ul style="list-style-type: none"> The current status of the order deposit (e.g., pending, paid, canceled). This helps track whether the deposit has been made, is awaiting payment, or has been canceled.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the deposit record was created. This helps track when the deposit was first initiated in the system.
	updated_at	<ul style="list-style-type: none"> The timestamp indicating the last time the deposit record was updated. This is used to track any changes made to the deposit after its creation.
ProductImages	product_image_id	<ul style="list-style-type: none"> A unique identifier for each product image. This serves as the primary key (PK) for the table, ensuring that each image record is distinct.
	imgUrl	<ul style="list-style-type: none"> The URL of the image associated with the product. This attribute stores the location of the image, which can be accessed or displayed within the system.

	type	<ul style="list-style-type: none"> The type or category of the image. This could indicate whether the image is a main product image, a thumbnail, or a secondary image showing different angles or details of the product.
	requested_product_id	<ul style="list-style-type: none"> The unique identifier of the requested product associated with the image. This foreign key links the image to a specific product request, helping organize and display the images relevant to the product.
OrderProgress	progress_id	<ul style="list-style-type: none"> A unique identifier for each progress record. This serves as the primary key (PK) for the table, ensuring that each progress entry is distinct.
	description	<ul style="list-style-type: none"> A textual description of the progress made on the order. This attribute provides information about the specific stage or update in the order's processing or fulfillment.
	created_by	<ul style="list-style-type: none"> The identifier of the user or entity who created or updated the progress record. This could refer to an employee, system, or other source responsible for tracking the order's progress.
	deleted_by	<ul style="list-style-type: none"> The identifier of the user or entity who deleted the progress record, if applicable. This attribute helps track who made changes to or removed progress entries from the system.
	updated_datetime	<ul style="list-style-type: none"> The timestamp indicating when the progress record was last updated. This helps track when the status or description of the order's progress was modified.
	status	<ul style="list-style-type: none"> The current status of the order's progress (e.g., in

		<p>progress, completed, canceled).</p> <ul style="list-style-type: none"> • This helps track the ongoing state of the order at any given point.
	order_id	<ul style="list-style-type: none"> • The unique identifier of the order associated with the progress record. • This foreign key links the progress entry to a specific order in the system.
	guarantee_order_id	<ul style="list-style-type: none"> • The unique identifier of the guarantee order associated with the progress record. • This foreign key links the progress entry to a specific guarantee order, if applicable.
QuotationDetail	quotation_detail_id	<ul style="list-style-type: none"> • A unique identifier for each quotation detail record. • This serves as the primary key (PK) for the table, ensuring that each quotation entry is distinct.
	requested_product_id	<ul style="list-style-type: none"> • The unique identifier of the requested product associated with the quotation detail. • This foreign key links the quotation detail to a specific product that has been requested.
	cost_type	<ul style="list-style-type: none"> • The type of cost associated with the quotation detail (e.g., labor, material, service). • This attribute helps categorize the cost and provides clarity on what the quoted amount represents.
	cost_amount	<ul style="list-style-type: none"> • The amount for the cost type in the quotation. • This attribute specifies the monetary value associated with the quoted cost for the requested product.
	quantity_required	<ul style="list-style-type: none"> • The quantity of the requested product required for the quotation. • This attribute indicates how many units of the product are being quoted for.

	guarantee_order_id	<ul style="list-style-type: none"> The unique identifier of the guarantee order associated with the quotation detail. This foreign key links the quotation detail to a specific guarantee order, if applicable.
Shipment	shipment_id	<ul style="list-style-type: none"> A unique identifier for each shipment record. This serves as the primary key (PK) for the table, ensuring that each shipment entry is distinct.
	to_address	<ul style="list-style-type: none"> The destination address where the shipment is being sent. This attribute specifies the address of the recipient or the delivery location for the shipment.
	shipping_unit	<ul style="list-style-type: none"> The unit responsible for handling the shipment, such as a shipping company or carrier. This attribute helps identify the organization or service managing the delivery.
	total_fee	<ul style="list-style-type: none"> The total cost of shipping the product. This attribute represents the full fee charged for the shipment, which may include packaging, handling, and delivery charges.
	order_code	<ul style="list-style-type: none"> The unique identifier associated with the order being shipped. This attribute helps link the shipment to a specific order in the system.
	ship_type	<ul style="list-style-type: none"> The type of shipping used for the shipment (e.g., standard, expedited, overnight). This attribute defines the shipping method chosen for delivery.
	order_id	<ul style="list-style-type: none"> The unique identifier of the order associated with the shipment. This foreign key links the shipment to a specific order in the system.

	from_address	<ul style="list-style-type: none"> The origin address from where the shipment is being sent. This attribute specifies the address of the sender or the warehouse from which the shipment is dispatched.
	guarantee_order_id	<ul style="list-style-type: none"> The unique identifier of the guarantee order associated with the shipment, if applicable. This foreign key links the shipment to a specific guarantee order, in case the shipment is related to a warranty or service issue.
Transaction	transaction_id	<ul style="list-style-type: none"> A unique identifier for each transaction record. This serves as the primary key (PK) for the table, ensuring that each transaction entry is distinct.
	user_id	<ul style="list-style-type: none"> The unique identifier of the user associated with the transaction. This foreign key links the transaction to a specific user in the system.
	transaction_type	<ul style="list-style-type: none"> The type of transaction being performed (e.g., debit, credit, refund). This attribute helps categorize the transaction and provides clarity on whether money is being added or deducted.
	payment_for	<ul style="list-style-type: none"> The purpose or reason for the transaction (e.g., payment for order, deposit, service fee). This attribute clarifies what the transaction is paying for.
	order_deposit_id	<ul style="list-style-type: none"> The unique identifier of the order deposit associated with the transaction. This foreign key links the transaction to a specific deposit made for an order.
	status	<ul style="list-style-type: none"> The current status of the transaction (e.g., pending, completed, failed). This helps track whether the transaction has been

		successfully processed or is awaiting completion.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the transaction was created. This helps track when the transaction was initiated or processed in the system.
	description	<ul style="list-style-type: none"> A detailed textual description of the transaction. This attribute can provide additional context or notes related to the transaction, such as the purpose or any special instructions.
	wallet_id	<ul style="list-style-type: none"> The unique identifier of the wallet used in the transaction. This foreign key links the transaction to a specific user's wallet, indicating which wallet the transaction affected.
	amount	<ul style="list-style-type: none"> The amount of money involved in the transaction. This attribute represents the value of the transaction, either as a debit or credit to the user's account or wallet.
Complaint	complaint_id	<ul style="list-style-type: none"> A unique identifier for each complaint record. This serves as the primary key (PK) for the table, ensuring that each complaint entry is distinct.
	order_id	<ul style="list-style-type: none"> The unique identifier of the order associated with the complaint. This foreign key links the complaint to a specific order in the system, allowing it to be tracked in relation to the particular order.
	description	<ul style="list-style-type: none"> A detailed textual description of the complaint. This attribute outlines the nature of the complaint, including the issue or problem the customer has encountered.

	status	<ul style="list-style-type: none"> The current status of the complaint (e.g., pending, resolved, closed). This helps track the progress of the complaint resolution process.
	created_at	<ul style="list-style-type: none"> The timestamp indicating when the complaint was created. This helps track when the complaint was filed by the customer.
	woodworker_response	<ul style="list-style-type: none"> The response or feedback provided by the woodworker regarding the complaint. This attribute records how the woodworker addressed or resolved the issue raised by the customer.

Table 13 - ERD dictionary

4. Non-Functional Requirements

4.1 External Interfaces

4.1.1 User Interfaces

- All the texts, labels and should be written in Vietnamese.

4.2 Quality Attributes

4.2.1 Usability

- Customers and woodworkers should be able to become productive with the system after less than one week of using.

4.2.2 Reliability

- Admins and moderators will require some training to effectively use the system.

4.2.3 Performance

- For all common operations, the system should respond within 5 seconds, except for the "define pickup points" feature.

4.2.4 Design Constraints

- The application must operate using the Vietnamese language.

4.2.5 Security

- Each user role must have clearly defined permissions for interacting with the system.

5. Requirement Appendix

5.1 Business Rules

Code	Business definition
BR-01	Each email can only be used for one account.
BR-02	Woodworker registration requires admin approval.

BR-03	Only authenticated woodworkers can change the visibility of their profile, and visibility status must be consistent across all public search listings.
BR-04	Only in-warranty products can request guarantee
BR-05	Only completed orders are eligible
BR-06	Only supported image formats (JPG, PNG, JPEG; max 5MB each) are allowed for uploading.
BR-07	Expired warranties default to repair service
BR-08	A valid shipping address must be chosen
BR-09	Only orders with status “Đã hoàn tất” can be reviewed.
BR-10	Rating must be an integer between 1 and 5.
BR-11	All submitted reviews require staff approval before public display.
BR-12	Feedback can only be sent when an order is in “waiting for customer approval.”
BR-13	Customers may only view complaints they submitted.
BR-14	Woodworker sees only their own products.
BR-15	Price and stock quantity must be zero or positive numbers.
BR-16	User confirmation is mandatory before deletion
BR-17	Passwords must meet strength requirements (min. 5 characters).
BR-18	OTPs expire after 5 minutes or one use.
BR-19	Logging out must revoke all active tokens for the user.
BR-20	After logout, any attempt to access protected URLs redirects back to login.
BR-21	Logout action is audited in the security log.
BR-22	Phone number must follow the defined format (e.g. 10–11 digits).
BR-23	Full name cannot be empty and must be under 100 characters.
BR-24	New password must be at least 8 characters, contain uppercase, lowercase, and a digit.
BR-25	User cannot reuse any of their last 3 passwords.
BR-26	All other sessions are revoked upon a successful change.
BR-27	Only one valid OTP may exist per user at a time.
BR-28	Only products marked “Published” are visible.
BR-29	Default sort order is by product ID descending.
BR-30	Any applied filters must match valid categories.
BR-31	Only products flagged as “Published” are accessible to guests.
BR-32	Detail view must include name, images, price, description, and availability.
BR-33	Unpublished or deleted products cannot be viewed.
BR-34	Only woodworkers with “Approved” status appear.
BR-35	Default sort is by rating descending.
BR-36	Location and specialty filters (if used) must match valid system values.
BR-37	Detail view must include at least full name, rating, specialty, and location.
BR-38	Unapproved or deleted woodworkers are inaccessible and trigger the not-found flow.
BR-39	Only design ideas with status “Published” are shown.
BR-40	Default sort order is by creation date descending.
BR-41	Any applied category filters must match valid system values.
BR-42	Detail view must include title, description, images, and material list.
BR-43	Unpublished or removed ideas trigger the not-available flow.
BR-44	Only service orders belonging to the logged-in customer appear
BR-45	Default sort is by order date descending.

BR-46	Only orders with a valid status (Pending/Completed/Cancelled) are shown.
BR-47	Customers may view only their own service orders.
BR-48	All past status updates must be shown in chronological order.
BR-49	Attachments, if any, must be downloadable or viewable.
BR-50	Only guarantee orders belonging to the logged-in customer are shown
BR-51	Default sort order is by request date descending
BR-52	Only guarantee orders belonging to the logged-in customer are shown
BR-53	Default sort order is by request date descending
BR-54	Filter options (e.g. Pending, In Progress, Completed) must match the system's defined statuses
BR-55	Customers may view only their own guarantee/repair orders.
BR-56	All status updates must be shown in chronological order.
BR-57	Attachments (photos, PDFs) must be downloadable or viewable.
BR-58	Only addresses belonging to the logged-in customer are listed.
BR-59	The default (primary) address appears at the top.
BR-60	All addresses must include complete recipient, phone, and location details.
BR-61	Each customer may save up to 3 addresses.
BR-62	Every address must include recipient name, valid phone number, and a complete location.
BR-63	One address must be marked as the default before checkout.
BR-64	Address changes take effect immediately for future orders.
BR-65	Checkout requires at least one valid shipping address.
BR-66	Maximum quantity per product cannot exceed current stock.
BR-67	Maximum of 4 custom products per service request.
BR-68	All dimensions must be positive numbers.
BR-69	Items that have been requested with orders cannot be deleted.
BR-70	Only the logged-in woodworker can do action.
BR-71	Only the logged-in customer can do action.
BR-72	The system supports multiple deposit methods, including bank transfer, credit card, and cash.
BR-73	The system supports multiple withdrawal methods, including bank transfer.
BR-74	The performance list is updated in real-time based on the latest data.
BR-75	The list is accessible only to authorized users, such as moderators.
BR-76	The detailed performance data is updated in real-time and available only for authorized users.
BR-77	The system must ensure the email content is accurate and aligns with the performance data.
BR-78	Emails are only sent to valid, active woodworker email addresses.
BR-79	Reviews are displayed in the order they were submitted.
BR-80	The staff member can approve, reject, or flag reviews based on company policies.
BR-81	Responses are accessible based on the review they belong to.
BR-82	Responses must adhere to company guidelines.

BR-83	Complaints must be reviewed within a certain time frame.
BR-84	The complaint details are accessible only to authorized staff members.
BR-85	Complaints should be resolved or escalated within a defined time frame.
BR-86	Transactions should be displayed in reverse chronological order (latest first).
BR-87	Transaction details are accessible only to authorized users (e.g., admins).
BR-88	Charts should reflect real-time data.
BR-89	Registration details are displayed in the order they were submitted.
BR-90	The woodworker's registration details are only accessible to authorized users (admins).
BR-91	Registrations must be approved or rejected within a certain timeframe.
BR-92	Configurations are displayed in the order they were set.
BR-93	Only authorized users can modify configurations.

Table 14 - Business Rules

5.2 Application Message List

#	Message code	Message type	Context	Content
1	MSG01	Toast message	Successful action	Thành công *
2	MSG02	Toast message	Failed action	Thất bại *
3	MSG03	Toast message	Warning action	Cảnh báo *
4	MSG04	Toast message	Inform action	Thông tin *
5	MSG05	Delete confirm message	Click delete record	Bạn có chắc muốn xóa *
6	MSG06	Success page	Payment success, order success	Thành công *

Table 15 - Application Message List

IV. Software Design Description

1. System Design

1.1 System Architecture

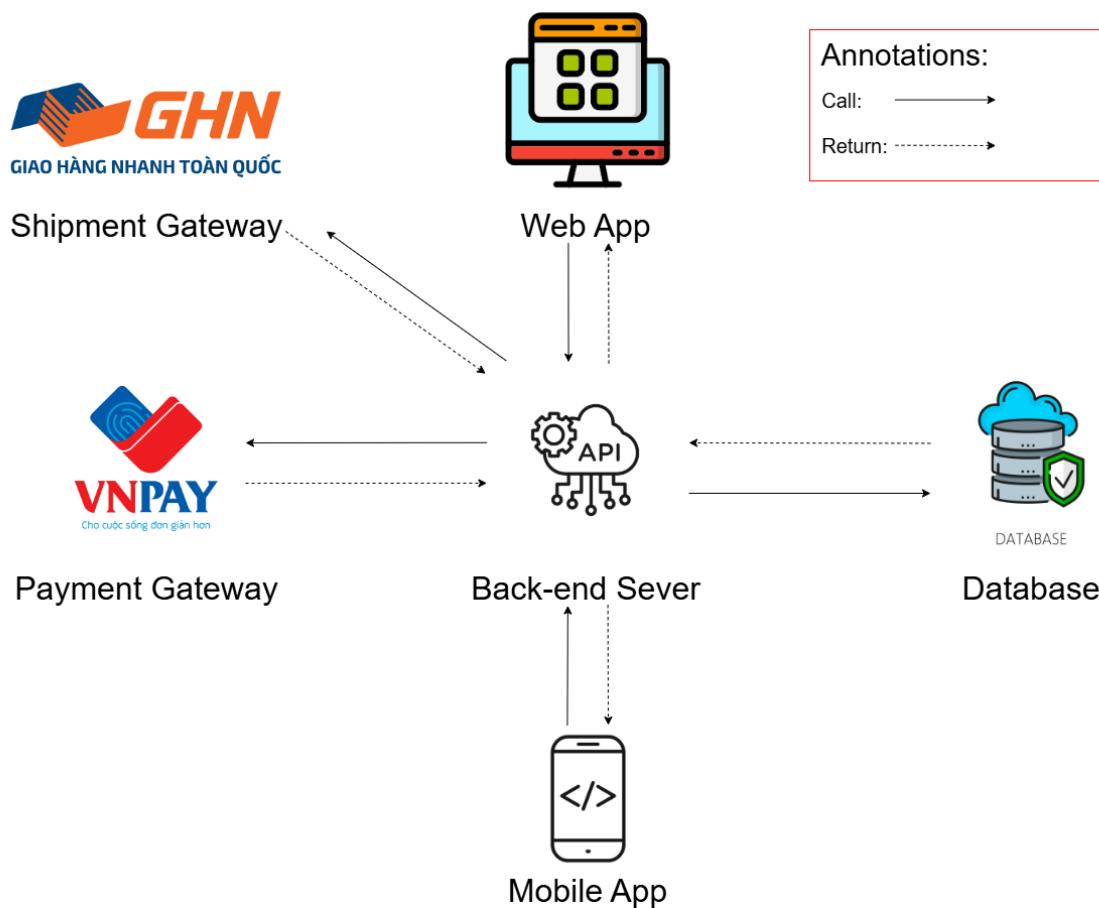


Figure 11 - System Architecture

The WDCRBP System includes a Back-end Server built with Java Spring Boot, which exposes APIs to support interactions between clients and third-party services. The Web App (built using React) is used by admins, moderators, customers, and woodworker for managing translation workflows. A Mobile App (built using React Native) is also available for users with similar functionalities, adapted for the mobile experience.

1.2 Package Diagram

1.2.1 Web API package diagram

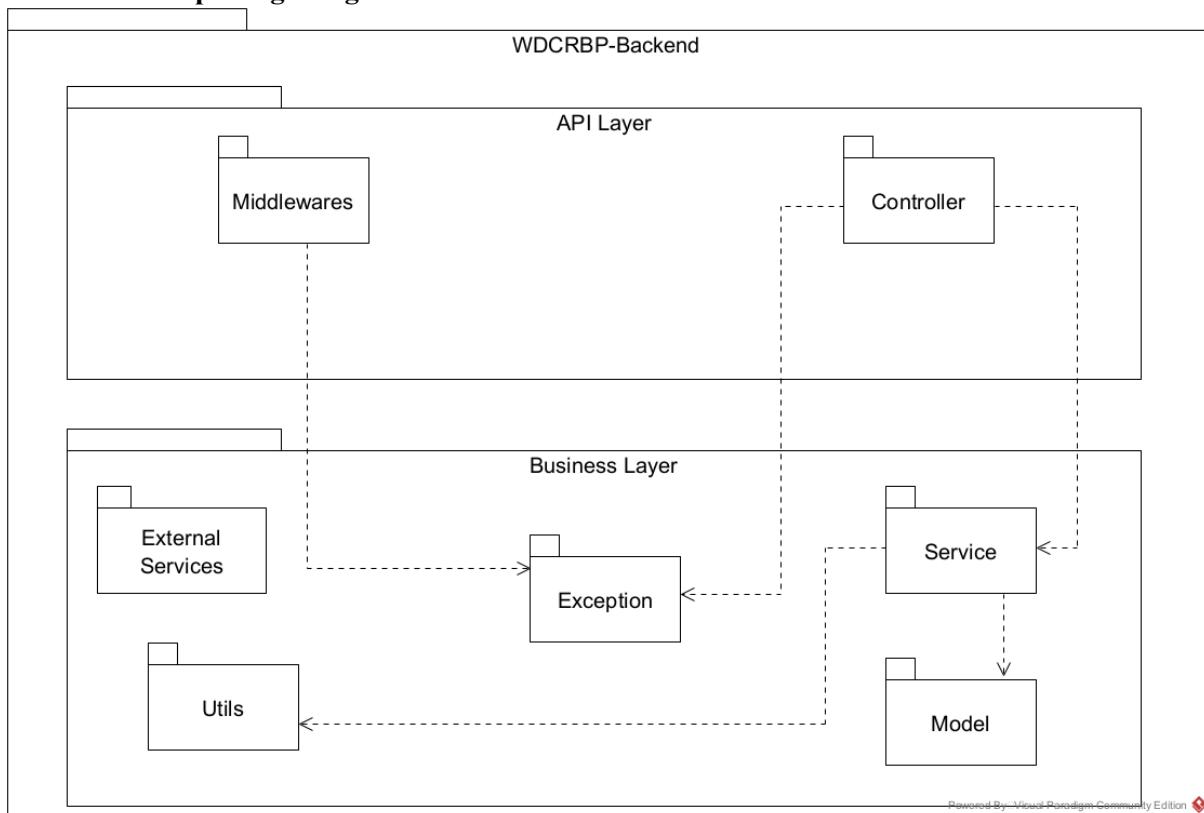


Figure 12 - Web API package diagram

1.2.2 Web application package diagram

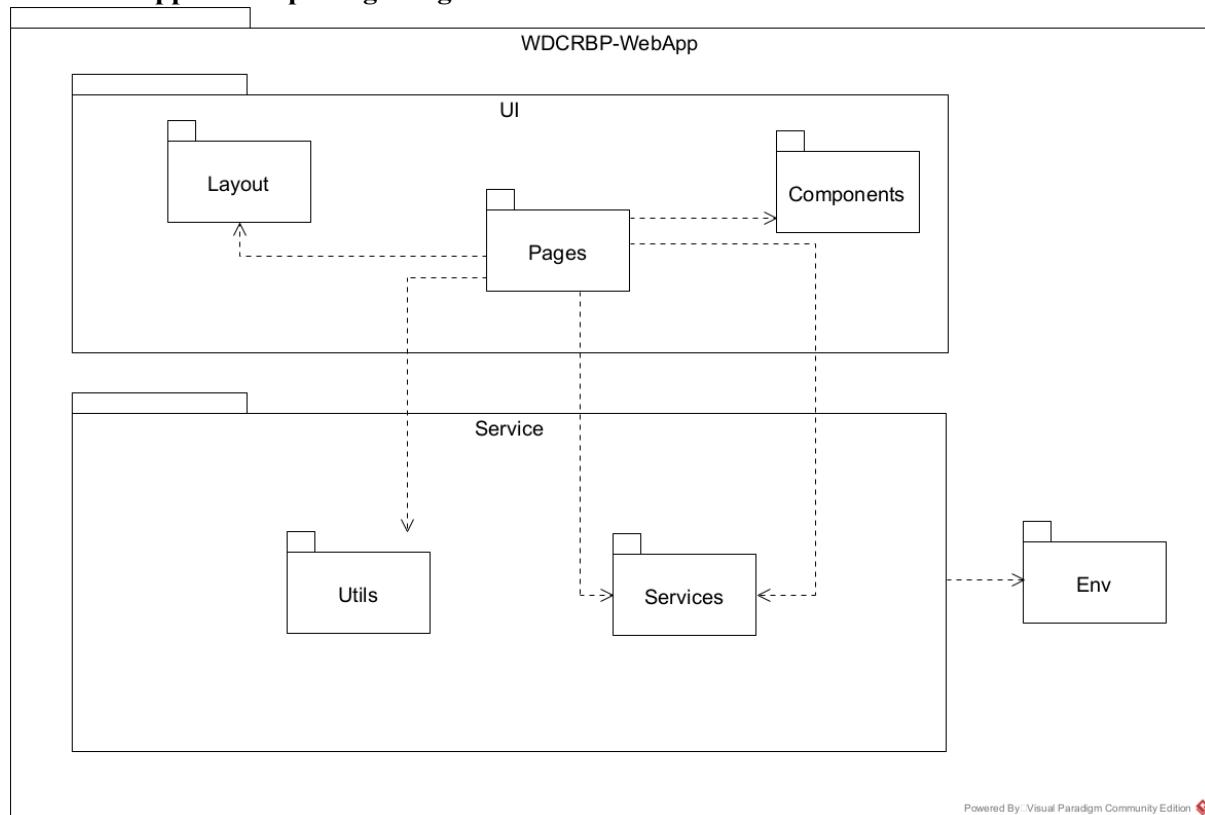


Figure 13 - Web application package diagram

1.2.3 Mobile application package diagram

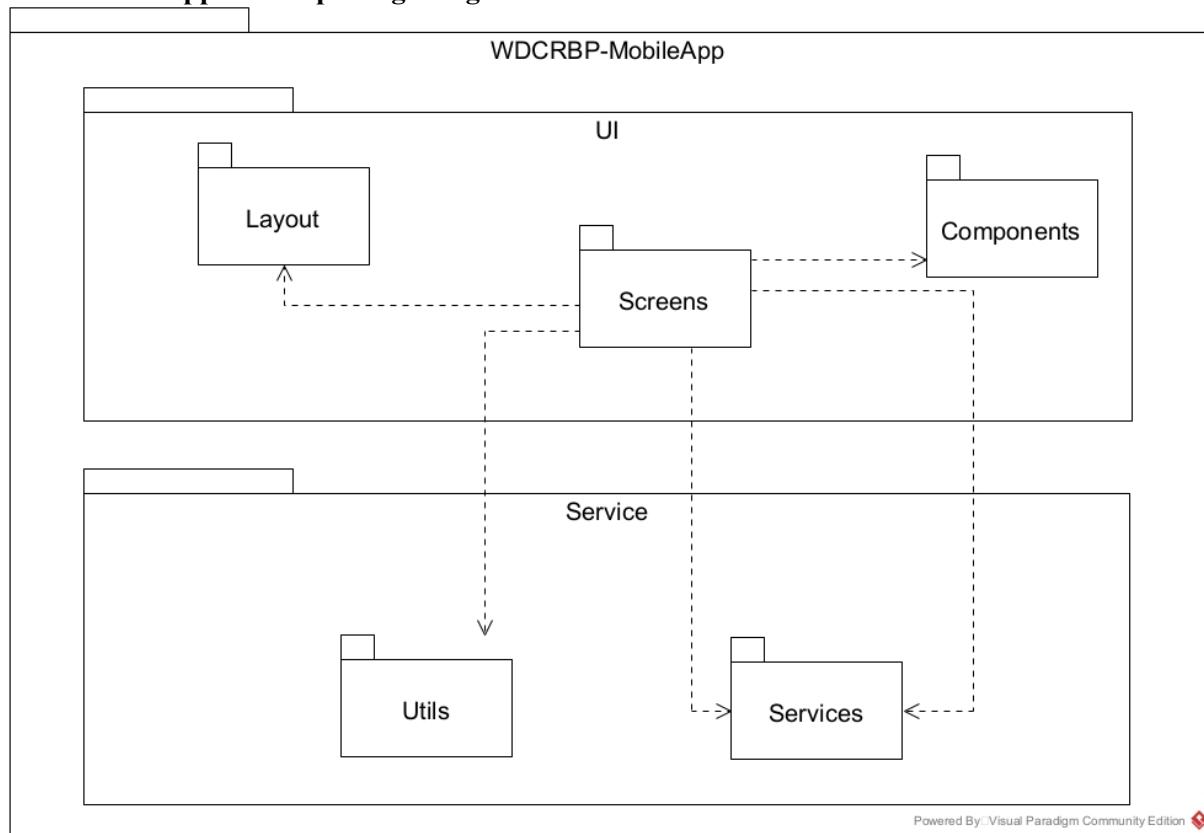


Figure 14 - Mobile application package diagram

2. Database Design

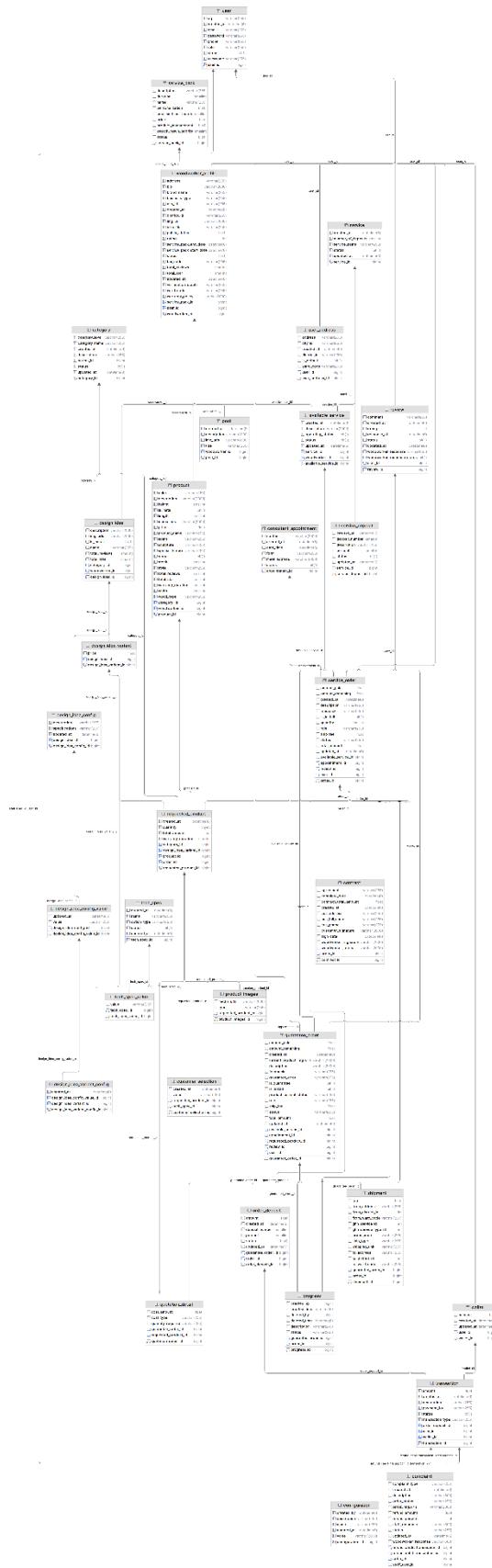


Figure 15 - Database design

Data Dictionary

Entity name	Attributes	Data type	Description
Category	category_id	bigint	Unique identifier for each category.
	category_level	varchar(255)	Level of the category in the hierarchy.
	category_name	varchar(255)	Name of the category.
	created_at	datetime(6)	Timestamp when the category was created.
	description	varchar(255)	Description of the category.
	parent_id	bigint	The parent category ID if applicable.
	status	bit	Current status of the category (active/inactive).
	updated_at	datetime(6)	Timestamp when the category was last updated.
Configuration	configuration_id	bigint	Unique identifier for each configuration.
	created_by	datetime(6)	Timestamp when the configuration was created.
	description	varchar(255)	Description of the configuration.
	name	varchar(255)	Name of the configuration.
	updated_at	datetime(6)	Timestamp when the configuration was last updated.
	value	varchar(2000)	Value of the configuration.
ConsultantAppointment	appointment_id	bigint	Unique identifier for each consultant appointment.
	content	varchar(2000)	Content or details of the appointment.
	created_at	datetime(6)	Timestamp when the appointment was created.
	date_time	datetime(6)	Date and time of the appointment.
	form	varchar(255)	Form or medium of the appointment (e.g., in-person, online).

	meet_address	varchar(2000)	The location where the meeting will take place.
	status	bit	Current status of the appointment.
Service	service_id	bigint	Unique identifier for each service.
	created_at	datetime(6)	Timestamp when the service was created.
	number_of_deposits	smallint	Number of deposits required for the service.
	service_name	varchar(255)	Name of the service.
	status	bit	Current status of the service (active/inactive).
	updated_at	datetime(6)	Timestamp when the service was last updated.
ServiceDeposit	service_deposit_id	bigint	Unique identifier for each service deposit.
	created_at	datetime(6)	Timestamp when the deposit record was created.
	deposit_number	smallint	Number of deposits required for the service.
	description	varchar(255)	Description of the deposit.
	percent	smallint	Percentage of the total amount for the deposit.
	status	bit	Current status of the deposit.
	updated_at	datetime(6)	Timestamp when the deposit record was last updated.
	service_id	bigint	The service ID related to the deposit.
ServicePack	service_pack_id	bigint	Unique identifier for each service pack.
	description	varchar(255)	Description of the service pack.
	duration	smallint	Duration for which the service pack is valid.
	name	varchar(255)	Name of the service pack.

	personalization	bit	Indicates if the service pack includes personalization.
	post_limit_per_month	smallint	Limit on the number of posts allowed per month under the pack.
	price	float	Price of the service pack.
	product_management	bit	Indicates if the service pack includes product management.
	search_result_priority	smallint	Priority of the service pack in search results.
TechSpec	tech_spec_id	bigint	Unique identifier for each technical specification.
	created_at	datetime(6)	Timestamp when the specification was created.
	name	varchar(255)	Name of the technical specification.
	option_type	varchar(255)	Type of option (e.g., dropdown, text input).
	status	bit	Current status of the specification (active/inactive).
	updated_at	datetime(6)	Timestamp when the specification was last updated.
TechSpecValue	tech_spec_value_id	bigint	Unique identifier for each technical specification value.
	value	varchar(255)	The value associated with the specification.
	tech_spec_id	bigint	The technical specification ID this value belongs to.
User	user_id	bigint	Unique identifier for each user.
	otp	varchar(255)	One-Time Password for user authentication.
	created_at	datetime(6)	Timestamp when the user was created.
	email	varchar(255)	User's email address.

	password	varchar(255)	User's password (hashed).
	phone	varchar(255)	User's phone number.
	role	varchar(255)	User's role (e.g., admin, customer).
	status	bit	Current status of the user (active/inactive).
	username	varchar(255)	User's chosen username.
Review	review_id	bigint	Unique identifier for each review.
	comment	varchar(255)	The comment or feedback provided by the reviewer.
	created_at	datetime(6)	Timestamp when the review was created.
	rating	int	Rating given by the reviewer.
	response_at	datetime(6)	Timestamp when the response to the review was made.
	status	bit	Current status of the review (approved/rejected).
	updated_at	datetime(6)	Timestamp when the review was last updated.
	woodworker_response	varchar(255)	Response provided by the woodworker.
	woodworker_response_status	bit	Status of the woodworker's response (e.g., pending, resolved).
	user_id	bigint	The ID of the user who wrote the review.
UserAddress	user_address_id	bigint	Unique identifier for each user address.
	address	varchar(255)	Full address of the user.
	city_id	varchar(255)	City associated with the user's address.
	created_at	datetime(6)	Timestamp when the user address was created.
	district_id	varchar(255)	District associated with the user's address.

	is_default	bit	Indicates if this is the user's default address.
	ward_code	varchar(255)	Ward or subdistrict code of the user's address.
	user_id	bigint	The ID of the user associated with the address.
Wallet	wallet_id	bigint	Unique identifier for each wallet.
	balance	float	The current balance in the wallet.
	created_at	datetime(6)	Timestamp when the wallet was created.
	updated_at	datetime(6)	Timestamp when the wallet was last updated.
	user_id	bigint	The ID of the user associated with the wallet.
WoodworkerProfile	woodworker_id	bigint	Unique identifier for each woodworker profile.
	address	varchar(255)	Address of the woodworker.
	bio	varchar(2000)	Bio or description of the woodworker.
	brand_name	varchar(255)	Name of the woodworker's brand.
	business_type	varchar(255)	Type of business the woodworker operates (e.g., individual, company).
	city_id	varchar(255)	City of the woodworker.
	created_at	datetime(6)	Timestamp when the woodworker profile was created.
	district_id	varchar(255)	District of the woodworker.
	img_url	varchar(2000)	URL to the profile image of the woodworker.
	no_order	varchar(255)	Number of orders completed by the woodworker.
	public_status	bit	Indicates whether the woodworker's profile is public.

	rating	int	Rating of the woodworker based on reviews.
	service_pack_end_date	datetime(6)	End date of the woodworker's service pack.
	service_pack_start_date	datetime(6)	Start date of the woodworker's service pack.
	status	bit	Current status of the woodworker's profile (active/inactive).
	tax_code	varchar(255)	Tax identification number of the woodworker.
	total_reviews	smallint	Total number of reviews received by the woodworker.
	total_star	smallint	Total star rating of the woodworker.
	updated_at	datetime(6)	Timestamp when the woodworker profile was last updated.
	verification_status	varchar(255)	Status of the woodworker's verification (e.g., verified, unverified).
	ward_code	varchar(255)	Ward code for the woodworker's address.
	warranty_policy	varchar(2000)	Warranty policy for the woodworker's products/services.
	service_pack_id	bigint	Service pack ID associated with the woodworker's profile.
	user_id	bigint	User ID of the associated woodworker profile.
AvailableService	available_service_id	bigint	Unique identifier for each available service.
	created_at	datetime(6)	Timestamp when the service was created.
	description	varchar(2000)	Detailed description of the available service.

	operating_status	bit	Current operating status of the service.
	status	bit	Current status of the service (active/inactive).
	updated_at	datetime(6)	Timestamp when the service details were last updated.
	service_id	bigint	Service ID associated with the available service.
	woodworker_id	bigint	Woodworker ID who offers the available service.
DesignIdea	design_idea_id	bigint	Unique identifier for each design idea.
	description	varchar(2000)	Detailed description of the design idea.
	img_urls	varchar(2000)	URLs of images related to the design idea.
	is_install	bit	Indicates if the design idea requires installation.
	name	varchar(255)	Name of the design idea.
	total_reviews	smallint	Total number of reviews for the design idea.
	total_star	smallint	Total star rating for the design idea.
	category_id	bigint	Category ID associated with the design idea.
	woodworker_id	bigint	Woodworker ID associated with the design idea.
DesignIdeaConfig	design_idea_config_id	bigint	Unique identifier for each design idea configuration.
	description	varchar(255)	Description of the design idea configuration.
	specifications	varchar(255)	Specifications related to the design idea configuration.
	updated_at	datetime(6)	Timestamp when the configuration was last updated.
	design_idea_id	bigint	Foreign key linking to the associated design idea.

DesignIdeaConfigValue	design_idea_config_value_id	bigint	Unique identifier for each design idea configuration value.
	updated_at	datetime(6)	Timestamp when the configuration value was last updated.
	value	varchar(255)	The value of the configuration.
	design_idea_config_id	bigint	Foreign key linking to the associated design idea configuration.
DesignIdeaVariant	design_idea_variant_id	bigint	Unique identifier for each design idea variant.
	price	float	Price of the design idea variant.
	design_idea_id	bigint	Foreign key linking to the associated design idea.
DesignIdeaVariantConfig	design_idea_variant_config_id	bigint	Unique identifier for each design idea variant configuration.
	updated_at	datetime(6)	Timestamp when the variant configuration was last updated.
	design_idea_config_value_id	bigint	Foreign key linking to the associated configuration value.
	design_idea_variant_id	bigint	Foreign key linking to the associated design idea variant.
Post	post_id	bigint	Unique identifier for each post.
	created_at	datetime(6)	Timestamp when the post was created.
	description	varchar(2000)	Description or content of the post.
	img_urls	varchar(2000)	URLs for images associated with the post.
	title	varchar(255)	Title of the post.
	woodworker_id	bigint	Foreign key linking to the woodworker who created the post.
Product	product_id	bigint	Unique identifier for each product.
	color	varchar(255)	Color of the product.

	description	varchar(2000)	Description of the product.
	height	smallint	Height of the product.
	is_install	bit	Indicates if installation is required.
	length	smallint	Length of the product.
	media_urls	varchar(2000)	URLs for media files (images/videos) of the product.
	price	float	Price of the product.
	product_name	varchar(255)	Name of the product.
	scent	varchar(255)	Scent of the product (if applicable).
	sculpture	varchar(255)	Indicates if the product is a sculpture.
	special_feature	varchar(255)	Special features of the product.
	status	bit	Current status of the product (active/inactive).
	stock	smallint	Number of items in stock.
	style	varchar(255)	Style of the product.
	total_reviews	smallint	Total number of reviews for the product.
	total_star	smallint	Average star rating of the product.
	warranty_duration	smallint	Duration of the product's warranty.
	width	smallint	Width of the product.
	wood_type	varchar(255)	Type of wood used for the product.
	category_id	bigint	Foreign key linking to the product category.
	woodworker_id	bigint	Foreign key linking to the woodworker.
ServiceOrder	order_id	bigint	Unique identifier for each service order.
	amount_paid	float	Amount already paid for the service order.

	amount_remaining	float	Amount remaining to be paid for the service order.
	created_at	datetime(6)	Timestamp when the service order was created.
	description	varchar(255)	Description of the service order.
	feedback	varchar(255)	Feedback or notes regarding the service order.
	is_install	bit	Indicates if installation is part of the service.
	quantity	smallint	Quantity of the service or products ordered.
	role	varchar(255)	Role of the person responsible for the order (e.g., customer).
	ship_fee	float	Shipping fee associated with the service order.
	status	varchar(255)	Current status of the service order.
	total_amount	float	Total amount for the service order.
	updated_at	datetime(6)	Timestamp when the service order was last updated.
	available_service_id	bigint	Foreign key linking to the available service.
	appointment_id	bigint	Foreign key linking to the associated appointment.
	review_id	bigint	Foreign key linking to the associated review.
	user_id	bigint	Foreign key linking to the user who made the order.
Contract	contract_id	bigint	Unique identifier for each contract.
	complete_date	datetime(6)	Date when the contract was completed.
	contract_total_amount	float	Total amount specified in the contract.
	created_at	datetime(6)	Timestamp when the contract was created.

	cus_address	varchar(255)	Customer's address in the contract.
	cus_full_name	varchar(255)	Customer's full name.
	cus_phone	varchar(255)	Customer's phone number.
	customer_signature	varchar(2000)	Signature of the customer.
	sign_date	datetime(6)	Date when the contract was signed.
	woodworker_signature	varchar(2000)	Signature of the woodworker.
	woodworker_terms	varchar(2000)	Terms agreed by the woodworker in the contract.
	order_id	bigint	Foreign key linking to the associated order.
	agreement	varchar(255)	General agreement terms.
RequestedProduct	requested_product_id	bigint	Unique identifier for each requested product.
	created_at	datetime(6)	Timestamp when the requested product was created.
	quantity	tinyint	Quantity of the requested product.
	total_amount	float	Total amount for the requested product.
	warranty_duration	smallint	Duration of the warranty for the requested product.
	category_id	bigint	Foreign key linking to the product category.
	design_idea_variant_id	bigint	Foreign key linking to the design idea variant.
	product_id	bigint	Foreign key linking to the specific product.
	order_id	bigint	Foreign key linking to the associated order.
CustomerSelection	customer_selection_id	bigint	Unique identifier for each customer selection.
	created_at	datetime(6)	Timestamp when the customer selection was created.

	value	varchar(255)	Value selected by the customer for a product option.
	requested_product_id	bigint	Foreign key linking to the requested product.
	tech_spec_id	bigint	Foreign key linking to the technical specification.
GuaranteeOrder	guarantee_order_id	bigint	Unique identifier for each guarantee order.
	amount_paid	float	Amount already paid for the guarantee order.
	amount_remaining	float	Amount remaining to be paid for the guarantee order.
	created_at	datetime(6)	Timestamp when the guarantee order was created.
	current_product_img_urls	varchar(2000)	URLs of images for the current product.
	description	varchar(2000)	Description of the guarantee order.
	feedback	varchar(255)	Feedback or comments related to the guarantee order.
	guarantee_error	varchar(255)	Any error encountered during the guarantee process.
	is_guarantee	bit	Indicates whether the product is under guarantee.
	is_install	bit	Indicates if installation is required for the guarantee.
	product_current_status	varchar(255)	Current status of the product under guarantee.
	role	varchar(255)	The role associated with the guarantee order.
	ship_fee	float	Shipping fee associated with the guarantee order.
	status	varchar(255)	Current status of the guarantee order.
	total_amount	float	Total amount for the guarantee order.

	updated_at	datetime(6)	Timestamp when the guarantee order was last updated.
	available_service_id	bigint	Foreign key linking to the available service.
	appointment_id	bigint	Foreign key linking to the associated appointment.
	requested_product_id	bigint	Foreign key linking to the requested product.
	review_id	bigint	Foreign key linking to the associated review.
	user_id	bigint	Foreign key linking to the user associated with the guarantee order.
OrderDeposit	order_deposit_id	bigint	Unique identifier for each order deposit.
	amount	float	The amount of the deposit for the order.
	created_at	datetime(6)	Timestamp when the deposit was created.
	deposit_number	smallint	The number of deposits required for the order.
	percent	smallint	The percentage of the total amount for the deposit.
	status	bit	Current status of the deposit (e.g., paid, pending).
	updated_at	datetime(6)	Timestamp when the deposit details were last updated.
	guarantee_order_id	bigint	Foreign key linking to the guarantee order.
	order_id	bigint	Foreign key linking to the service order.
ProductImages	product_images_id	bigint	Unique identifier for each product image.
	media_urls	varchar(2000)	URLs of the media files associated with the product.
	type	varchar(255)	Type of the image (e.g., main, thumbnail).

	requested_product_id	bigint	Foreign key linking to the requested product.
OrderProgress	progress_id	bigint	Unique identifier for each progress record.
	created_by	bigint	The user or entity who created the progress record.
	created_time	datetime(6)	Timestamp when the progress record was created.
	deleted_by	bigint	The user or entity who deleted the progress record.
	deleted_time	datetime(6)	Timestamp when the progress record was deleted.
	description	varchar(255)	Description of the progress made on the order.
	status	varchar(255)	Current status of the progress (e.g., in progress, completed).
	guarantee_order_id	bigint	Foreign key linking to the associated guarantee order.
	order_id	bigint	Foreign key linking to the associated service order.
QuotationDetail	quotation_detail_id	bigint	Unique identifier for each quotation detail.
	cost_amount	float	The cost amount for the quotation detail.
	cost_type	varchar(255)	The type of cost (e.g., labor, materials).
	quantity_required	varchar(255)	The quantity of the requested product for the quotation.
	guarantee_order_id	bigint	Foreign key linking to the associated guarantee order.
	requested_product_id	bigint	Foreign key linking to the requested product.
Shipment	shipment_id	bigint	Unique identifier for each shipment record.
	fee	float	The fee associated with the shipment.

	from_address	varchar(255)	The address from where the shipment is sent.
	from_district_id	int	The district ID from where the shipment is sent.
	from_ward_code	varchar(255)	The ward code from where the shipment is sent.
	ghn_service_id	int	The service ID associated with the shipment.
	ghn_service_type_id	int	The type ID of the shipping service.
	order_code	varchar(255)	The code associated with the order being shipped.
	ship_type	varchar(255)	The type of shipping method used (e.g., standard, expedited).
	shipping_unit	varchar(255)	The shipping unit or carrier handling the shipment.
	to_address	varchar(255)	The destination address for the shipment.
	to_district_id	int	The district ID of the destination.
	to_ward_code	varchar(255)	The ward code of the destination.
	guarantee_order_id	bigint	Foreign key linking to the associated guarantee order.
	order_id	bigint	Foreign key linking to the associated service order.
Transaction	transaction_id	bigint	Unique identifier for each transaction.
	amount	float	The amount of money involved in the transaction.
	created_at	datetime(6)	Timestamp when the transaction was created.
	description	varchar(255)	Description of the transaction.
	payment_for	varchar(255)	The purpose of the payment (e.g., service order).
	status	bit	Current status of the transaction (e.g., completed, failed).

	transaction_type	varchar(255)	Type of transaction (e.g., debit, credit).
	order_deposit_id	bigint	Foreign key linking to the order deposit associated with the transaction.
	user_id	bigint	Foreign key linking to the user who made the transaction.
	wallet_id	bigint	Foreign key linking to the wallet involved in the transaction.
Complaint	complaint_id	bigint	Unique identifier for each complaint record.
	complaint_type	varchar(255)	Type of complaint (e.g., service issue, product issue).
	created_at	datetime(6)	Timestamp when the complaint was created.
	description	varchar(500)	Detailed description of the complaint.
	proof_img_urls	varchar(2000)	URLs of images or proof related to the complaint.
	refund_amount	float	Amount of refund requested due to the complaint.
	staff_response	varchar(500)	Response from the staff regarding the complaint.
	status	varchar(255)	Current status of the complaint (e.g., pending, resolved).
	updated_at	datetime(6)	Timestamp when the complaint was last updated.
	woodworker_response	varchar(500)	Response from the woodworker regarding the complaint.
	refund_credit_transaction_id	bigint	Foreign key linking to the refund credit transaction.
	refund_debit_transaction_id	bigint	Foreign key linking to the refund debit transaction.
	order_id	bigint	Foreign key linking to the order associated with the complaint.

	staff_user_id	bigint	Foreign key linking to the staff who handled the complaint.
	order_status	varchar(255)	Status of the order associated with the complaint.
	refund_percent	int	Percentage of refund requested.

Table 16 - Database data dictionary

3. Detailed Design

3.1 Woodworker registration feature

3.1.1 Class diagram

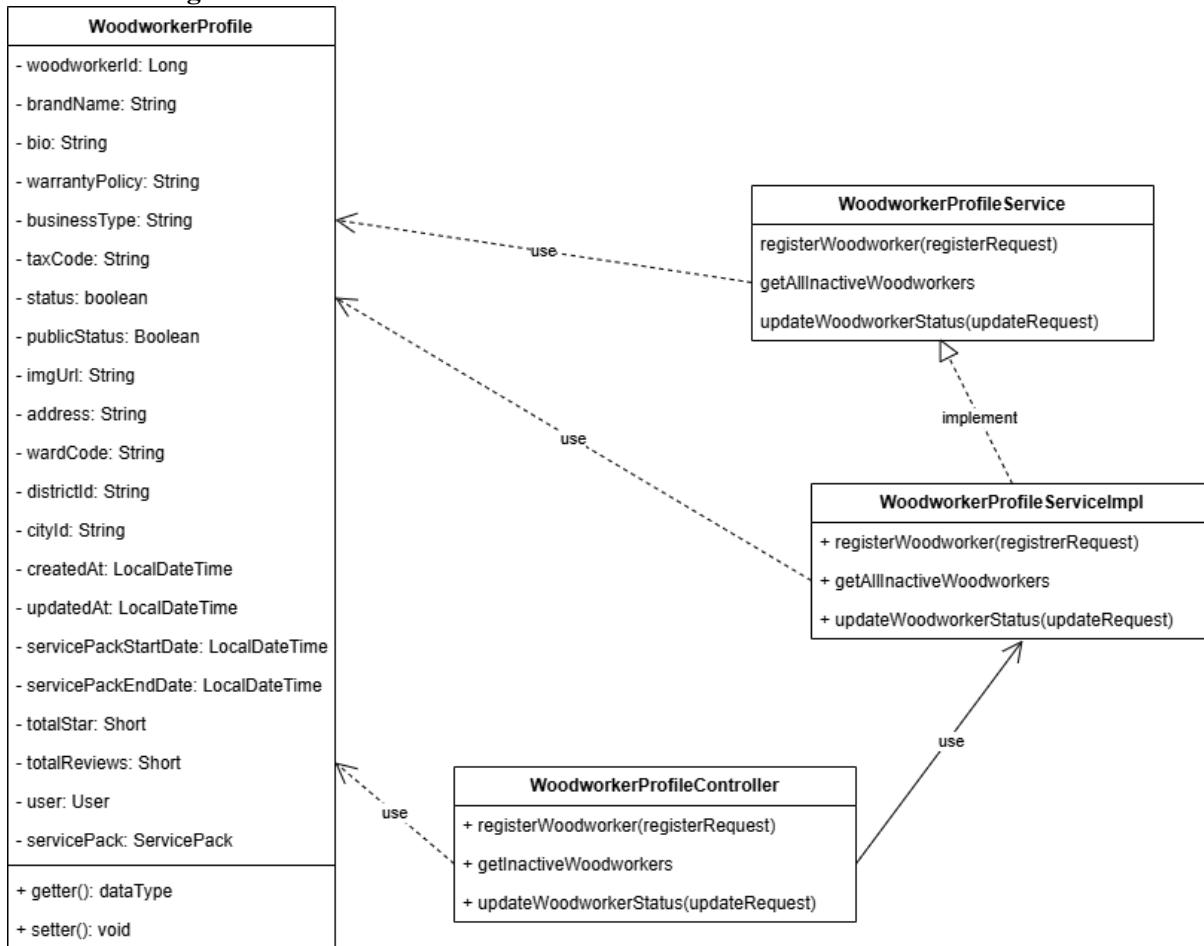


Figure 16 - Woodworker registration feature class diagram

3.1.2 Activity diagram

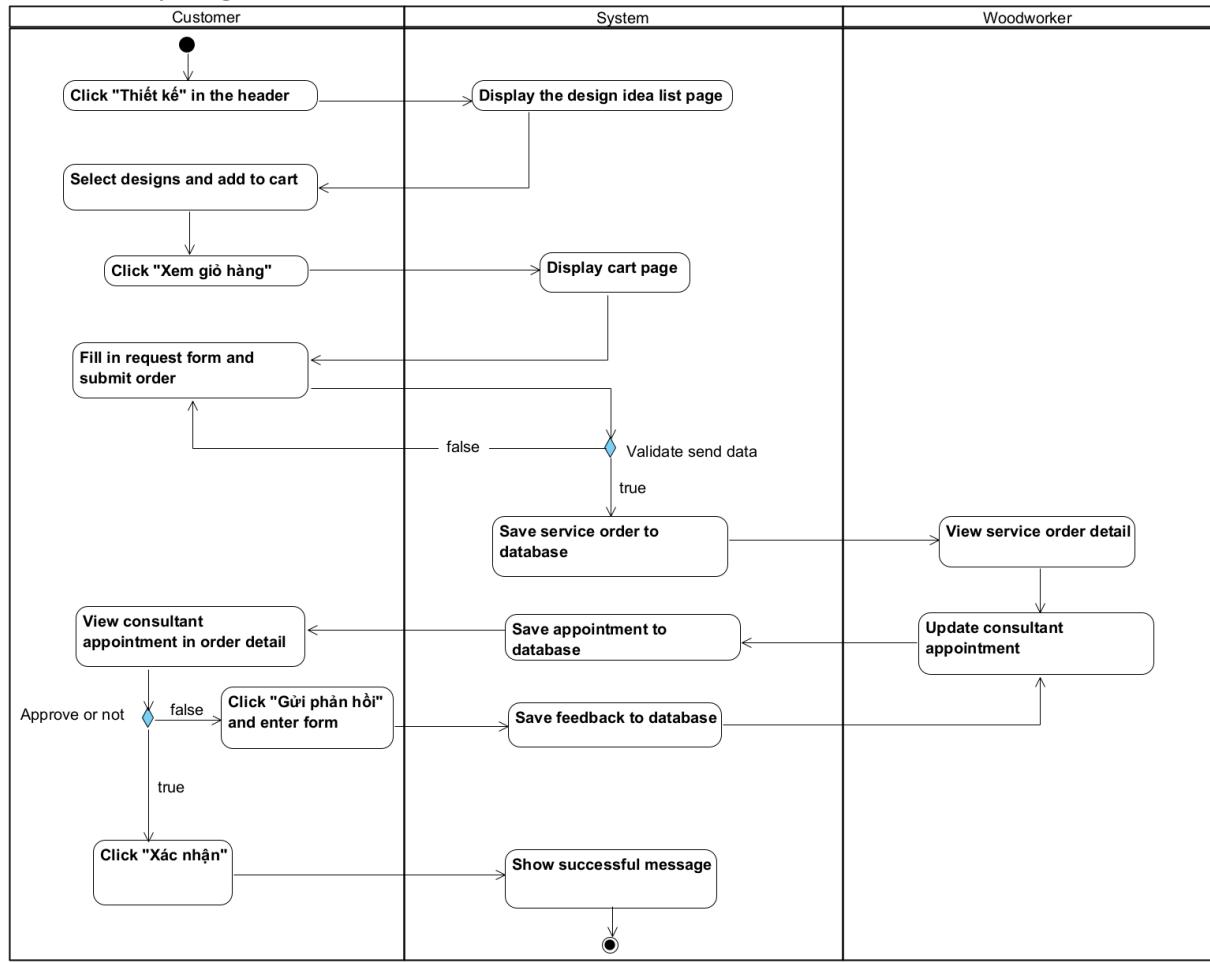


Figure 17 - Woodworker registration activity diagram

3.1.3 Sequence diagram

3.1.3.1 Register sequence diagram

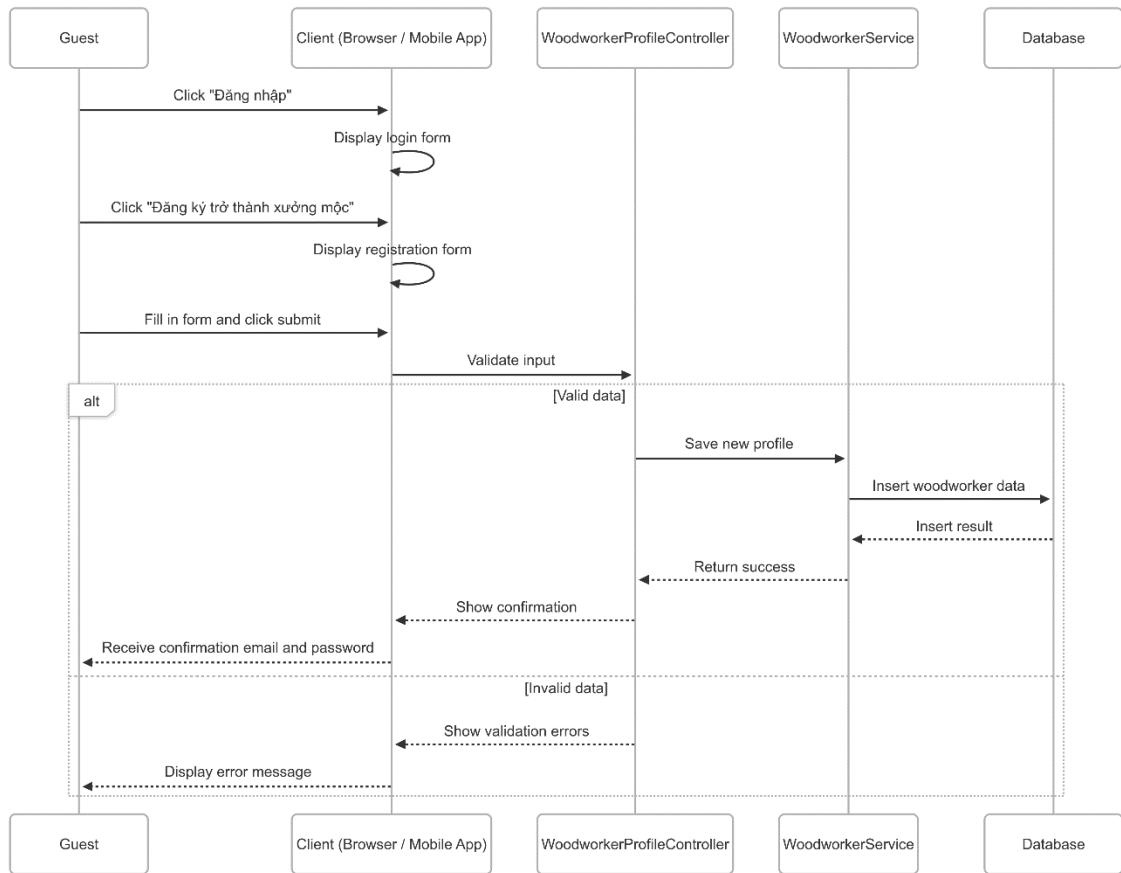


Figure 18 - Register sequence diagram

3.1.3.2 Approval sequence diagram

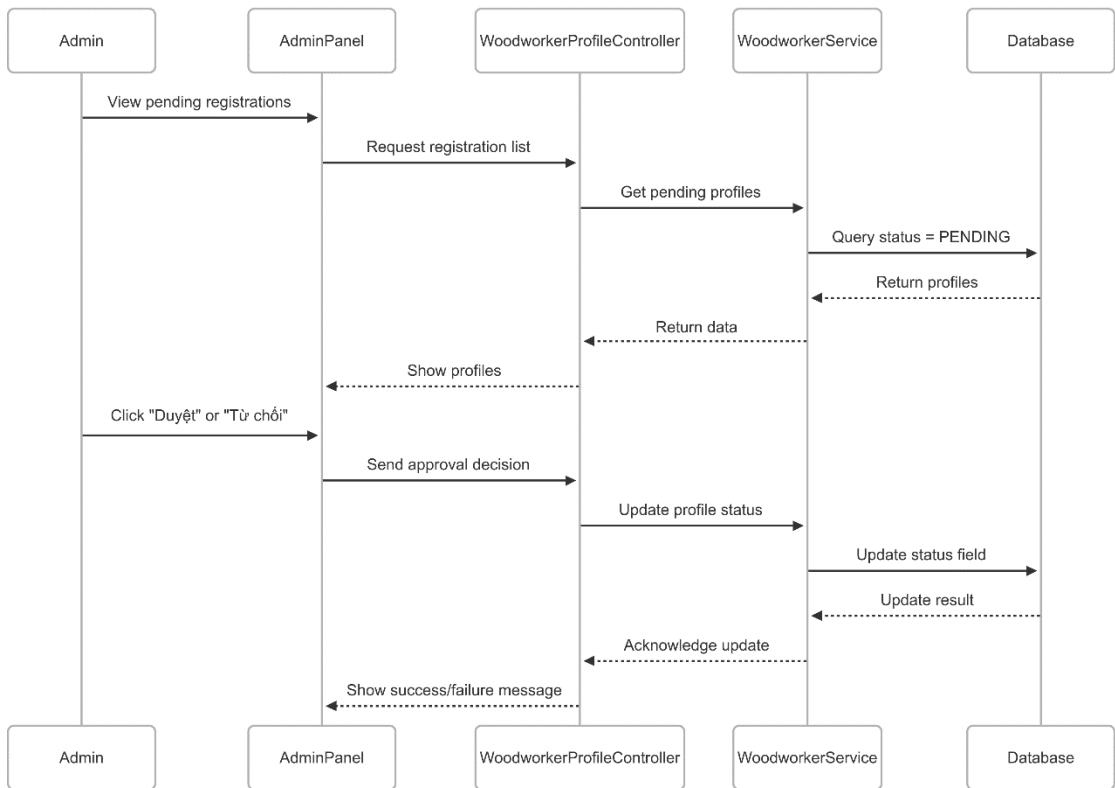


Figure 19 - Approval sequence diagram

3.2 Customization service booking feature

3.2.1 Class diagram

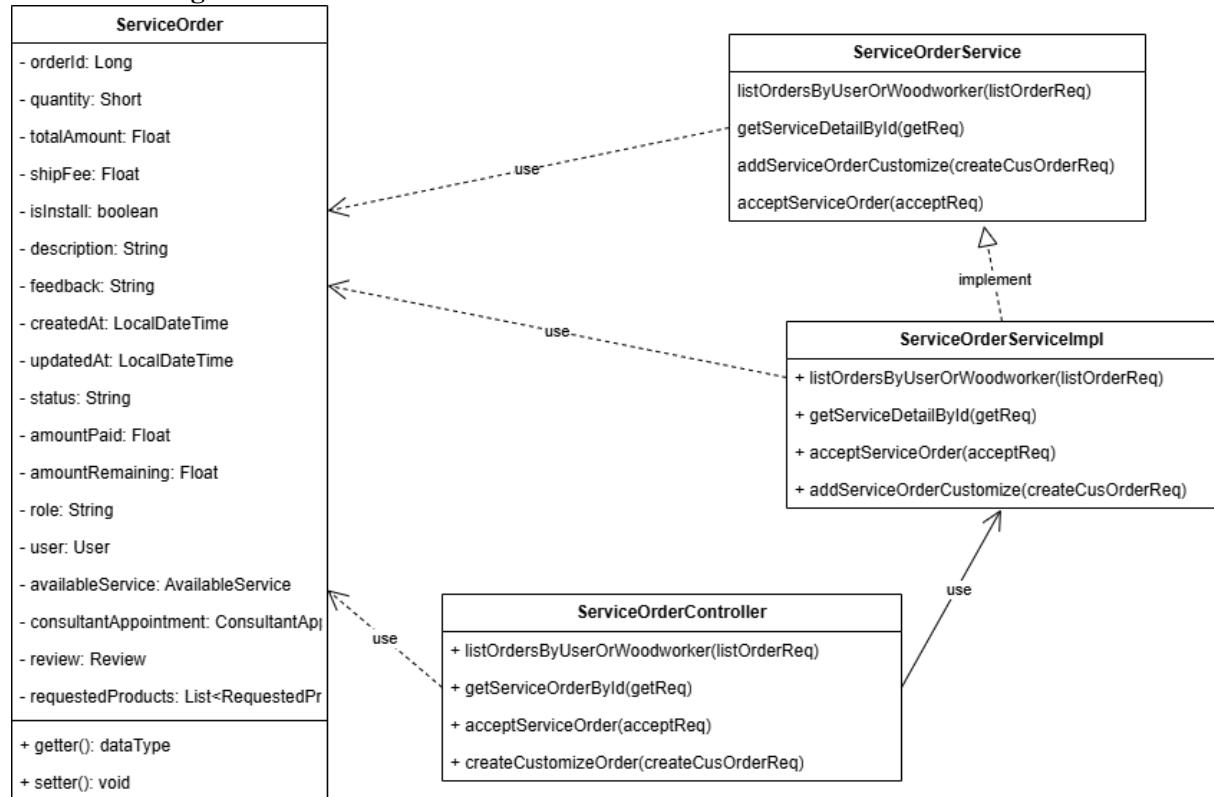


Figure 20 - Customization service booking feature class diagram

3.2.2 Activity diagram

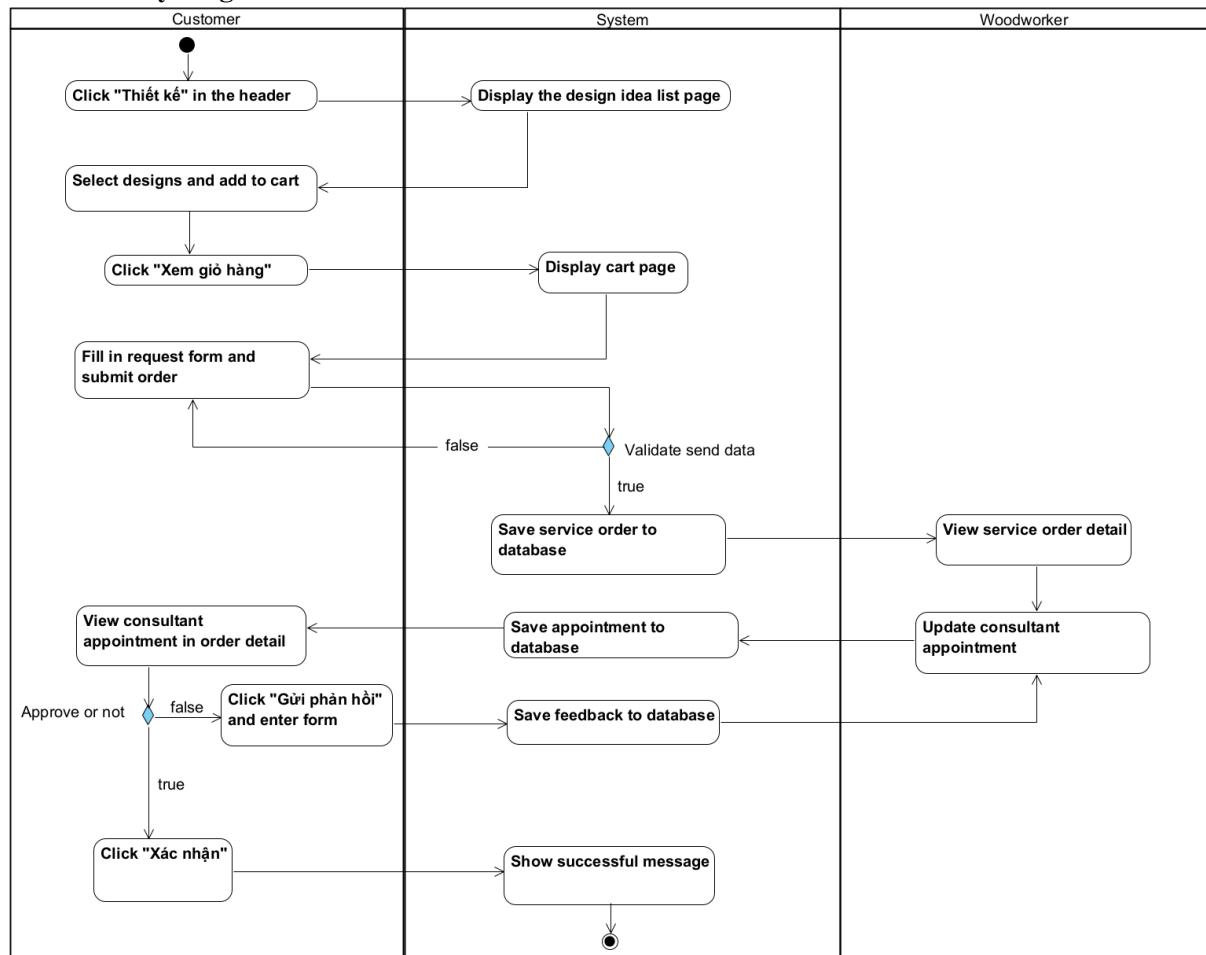


Figure 21 - Customization service booking feature activity diagram

3.2.3 Sequence diagram

3.2.3.1 Book customization service sequence diagram

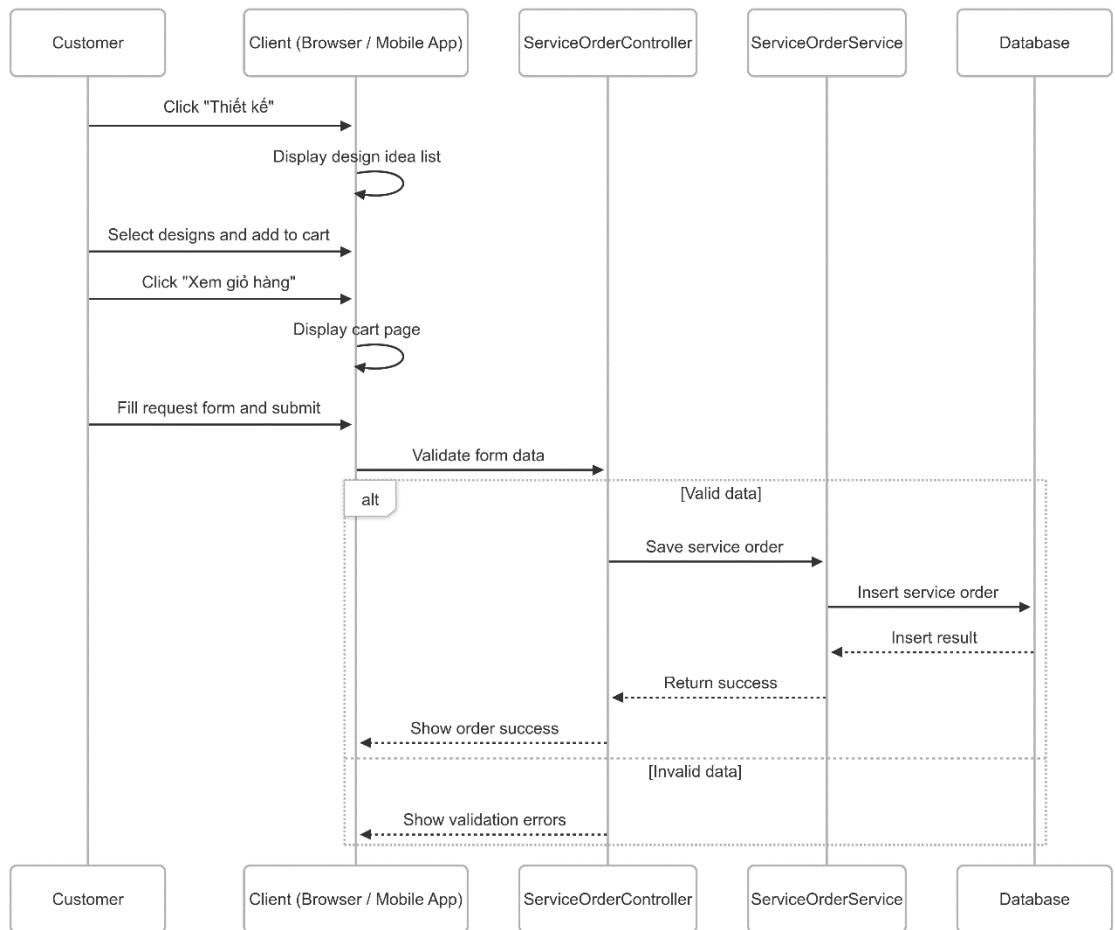


Figure 22 - Book customization service sequence diagram

3.2.3.2 Update consultant appointment for customization service sequence diagram

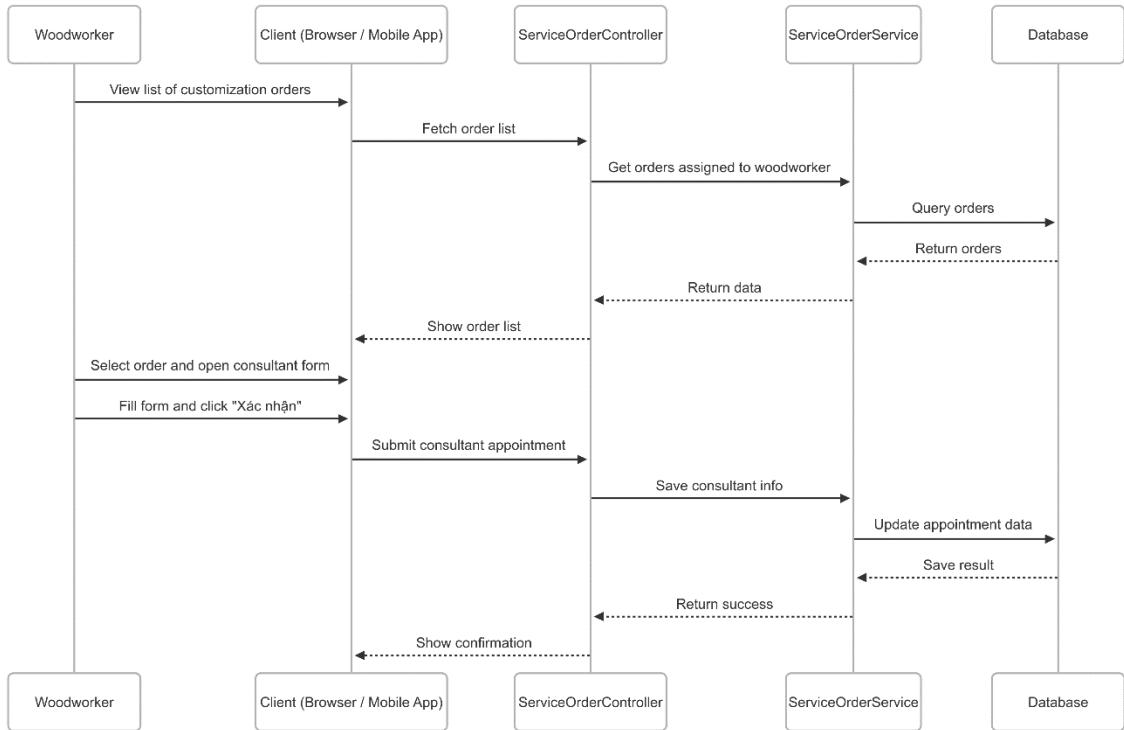


Figure 23 - Update consultant appointment for customization service sequence diagram

3.2.4 Customization order state machine diagram

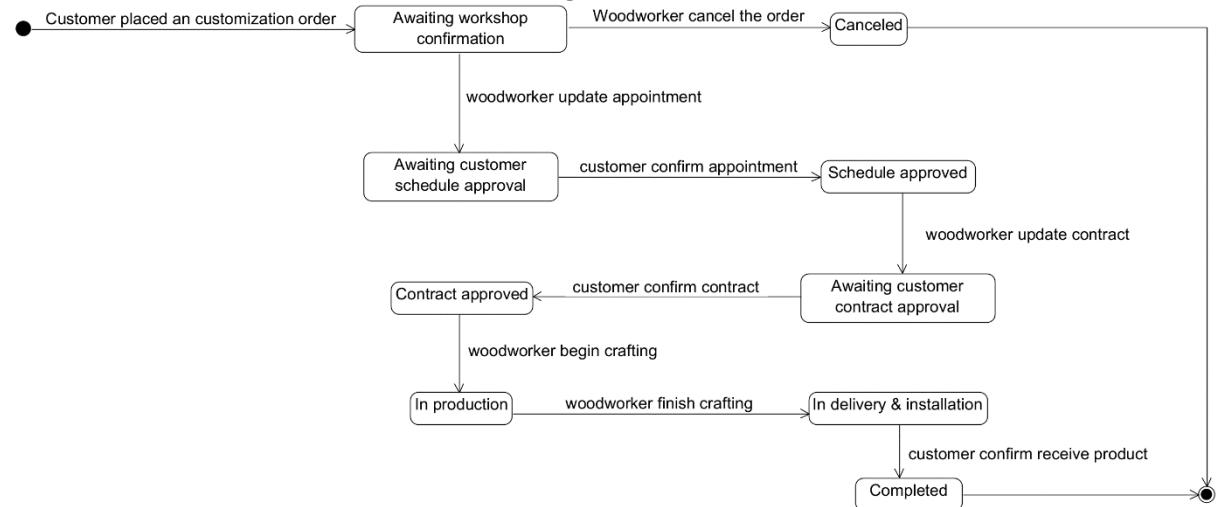


Figure 24 - Customization order state machine diagram

3.3 Personalization service booking feature

3.3.1 Class diagram

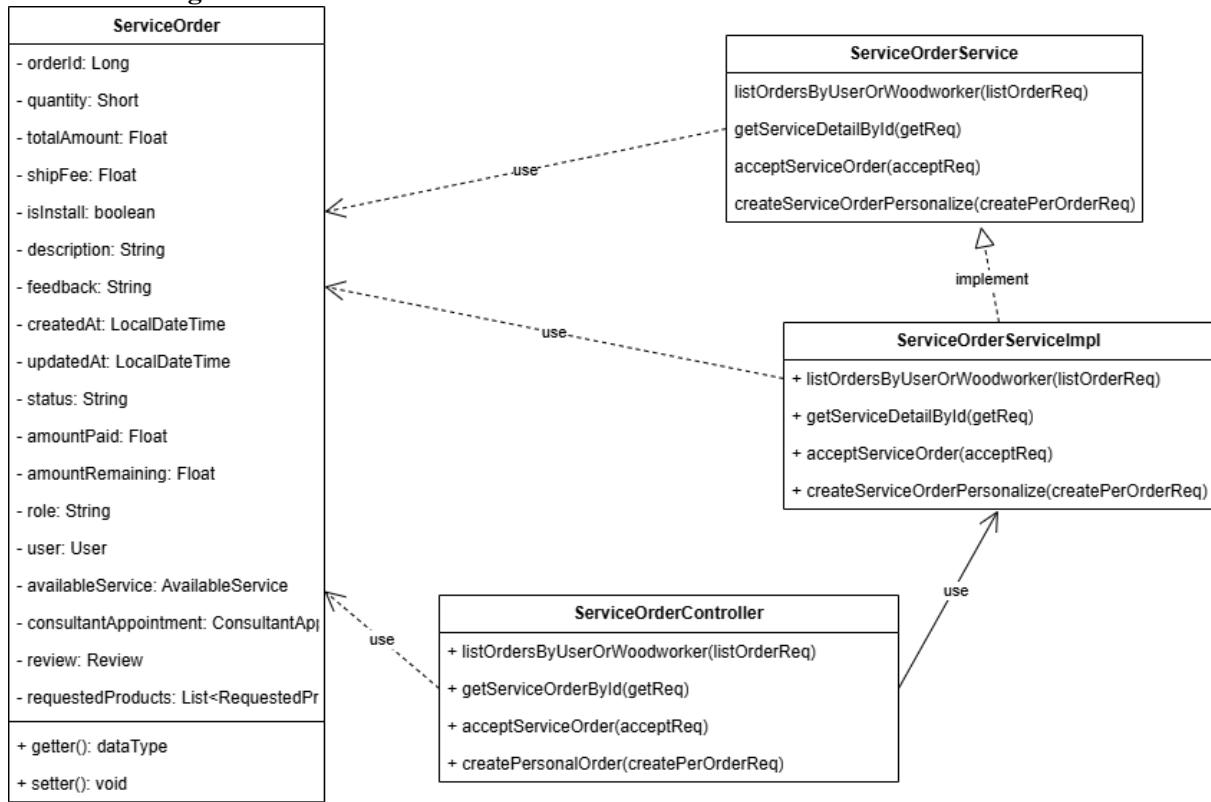


Figure 25 - Personalization service booking feature class diagram

3.3.2 Activity diagram

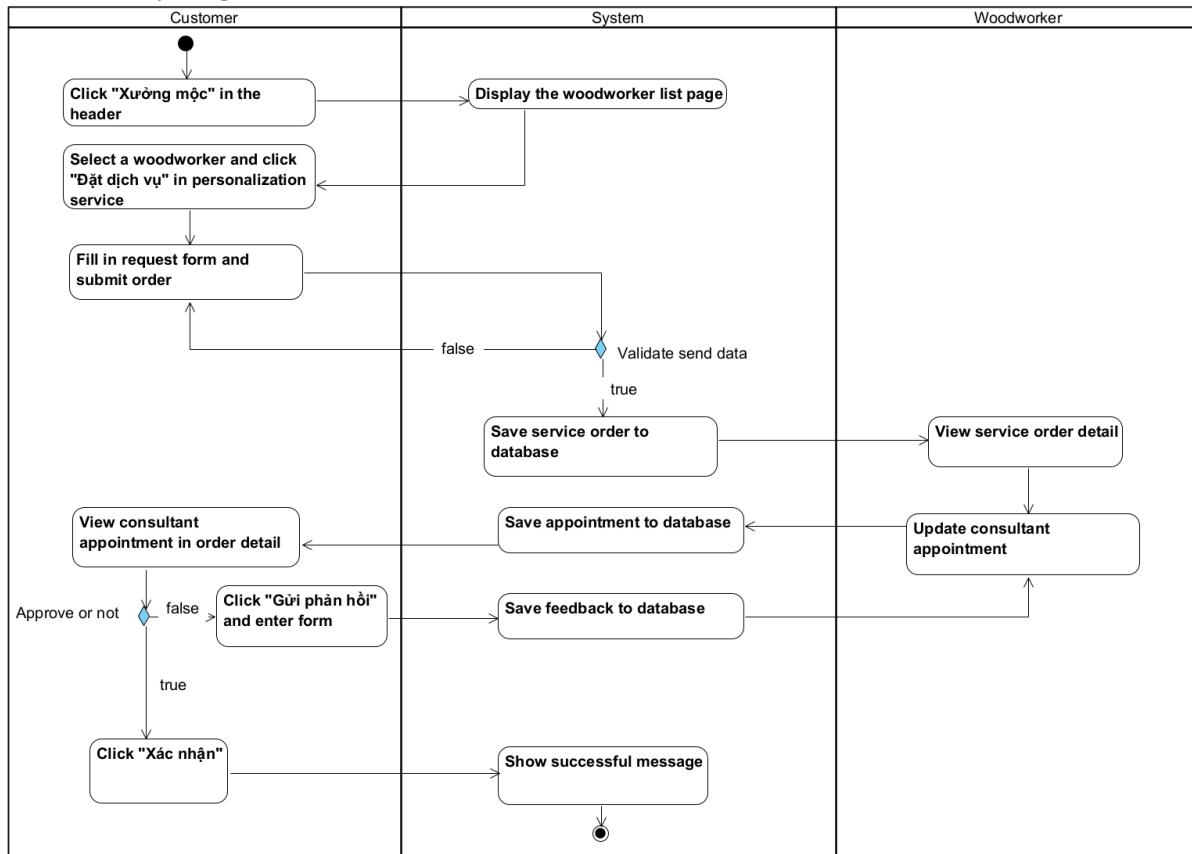


Figure 26 - Personalization service booking feature activity diagram

3.3.3 Sequence diagram

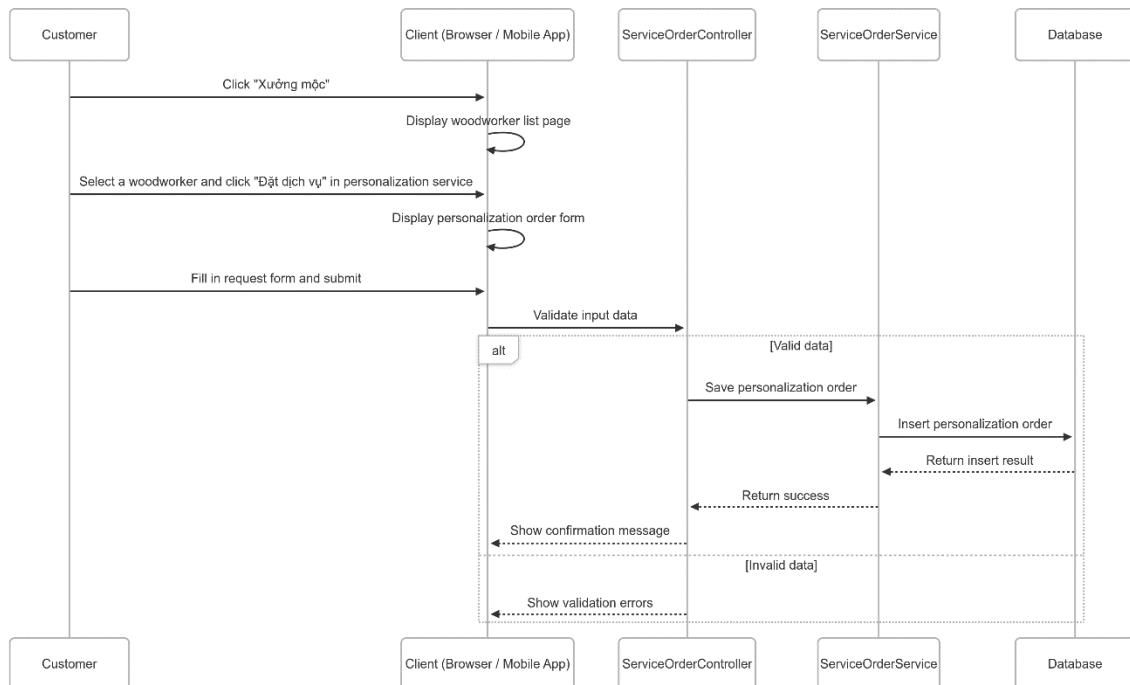


Figure 27 - Personalization service booking feature sequence diagram

3.3.4 Personalization order state machine diagram

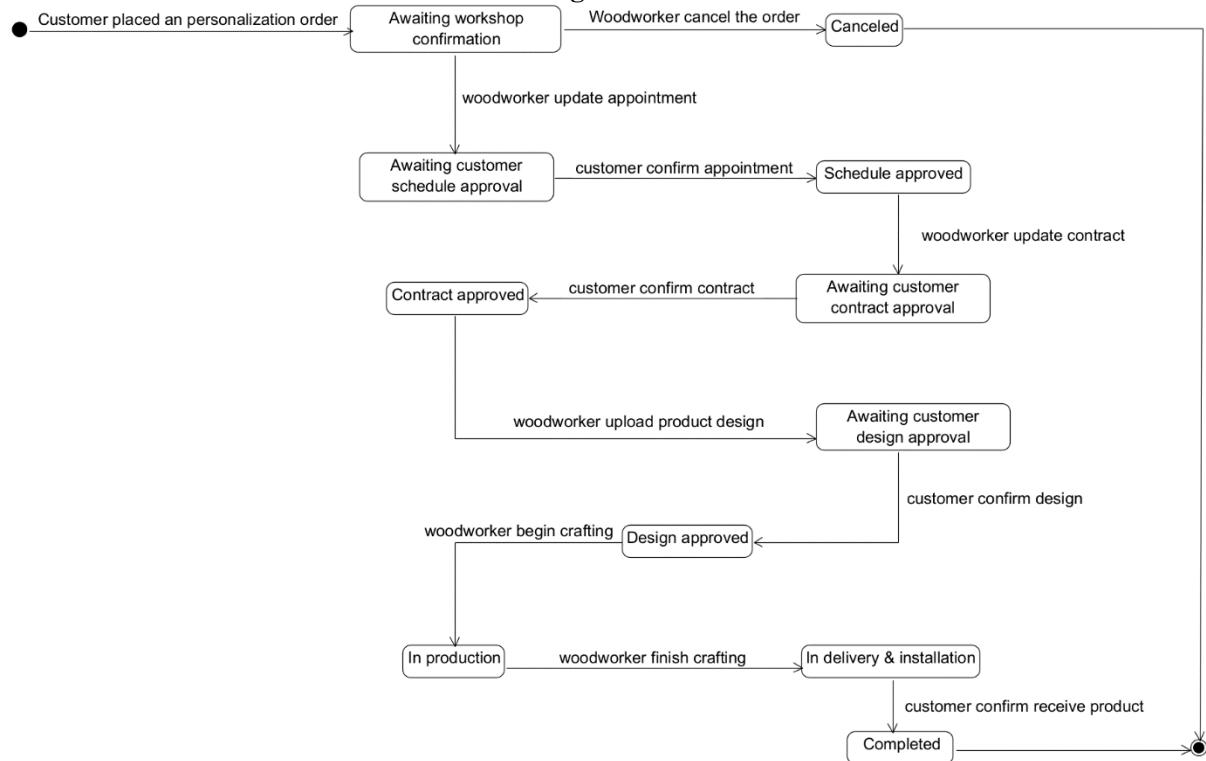


Figure 28 - Personalization order state machine diagram

3.4 Guarantee service booking feature

3.4.1 Class diagram

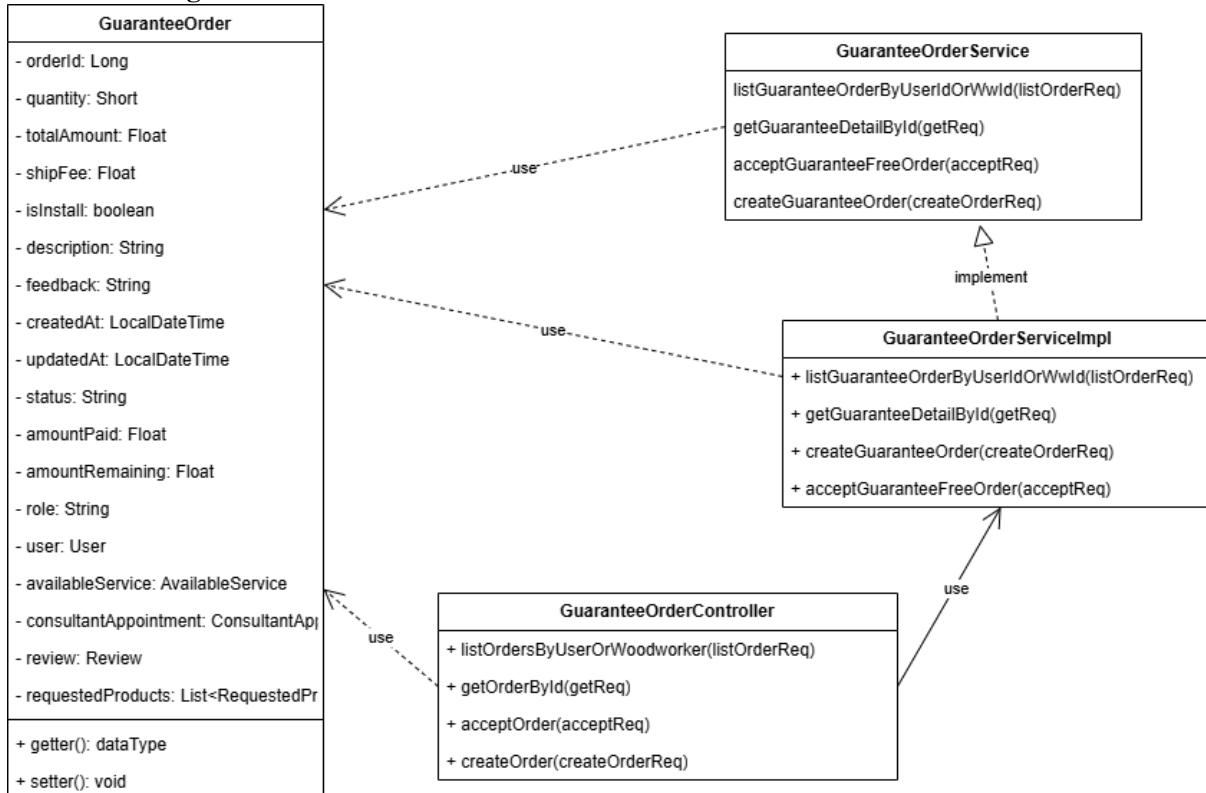


Figure 29 - Guarantee service booking feature class diagram

3.4.2 Activity diagram

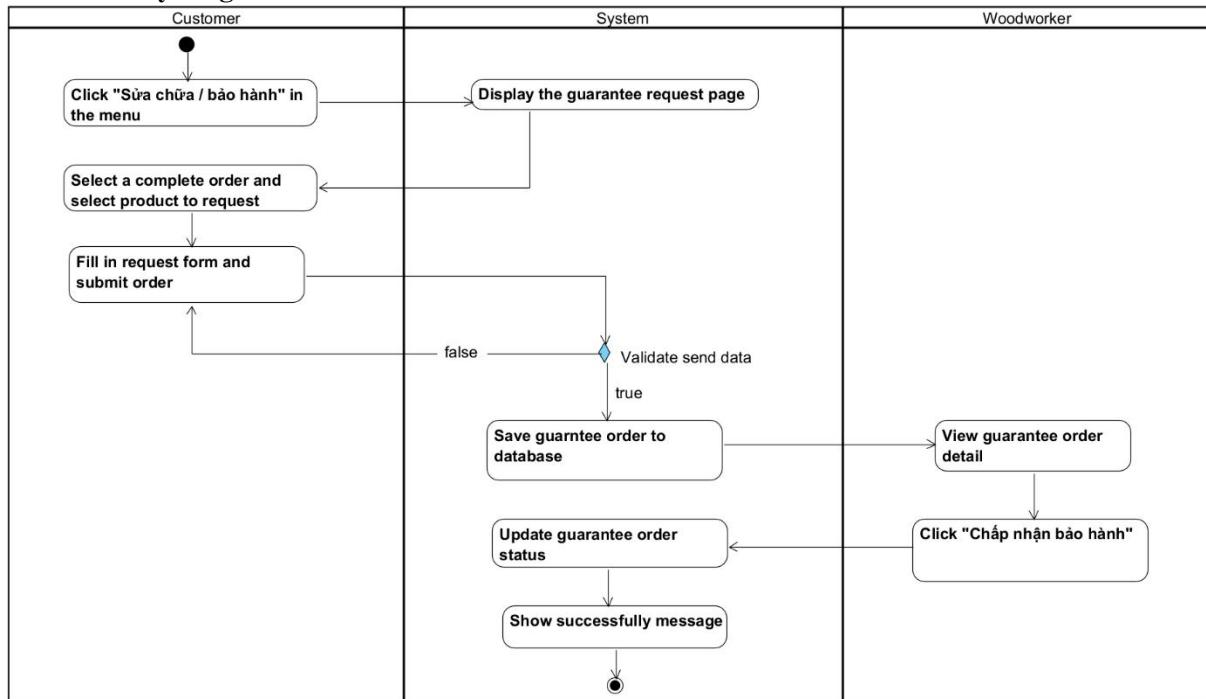


Figure 30 - Guarantee service booking feature activity diagram

3.4.3 Sequence diagram

3.4.3.1 Request guarantee order sequence diagram

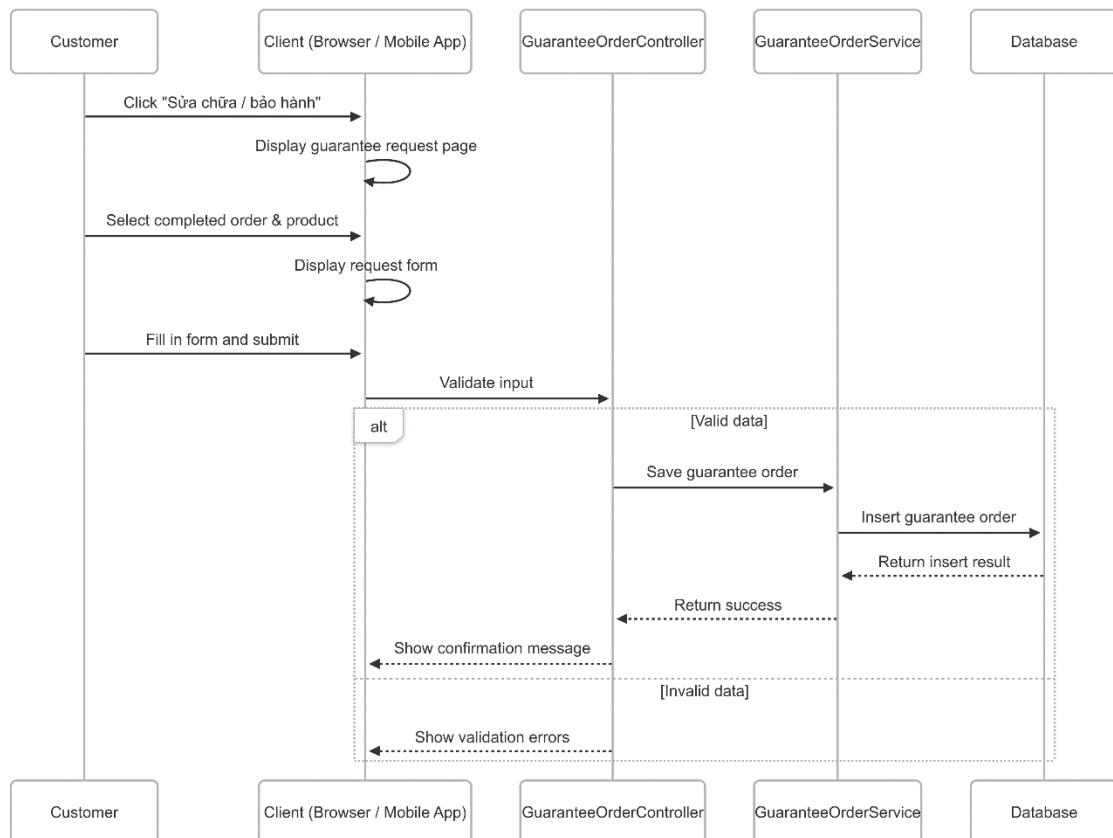


Figure 31 - Request guarantee order sequence diagram

3.4.3.2 Accept guarantee order sequence diagram

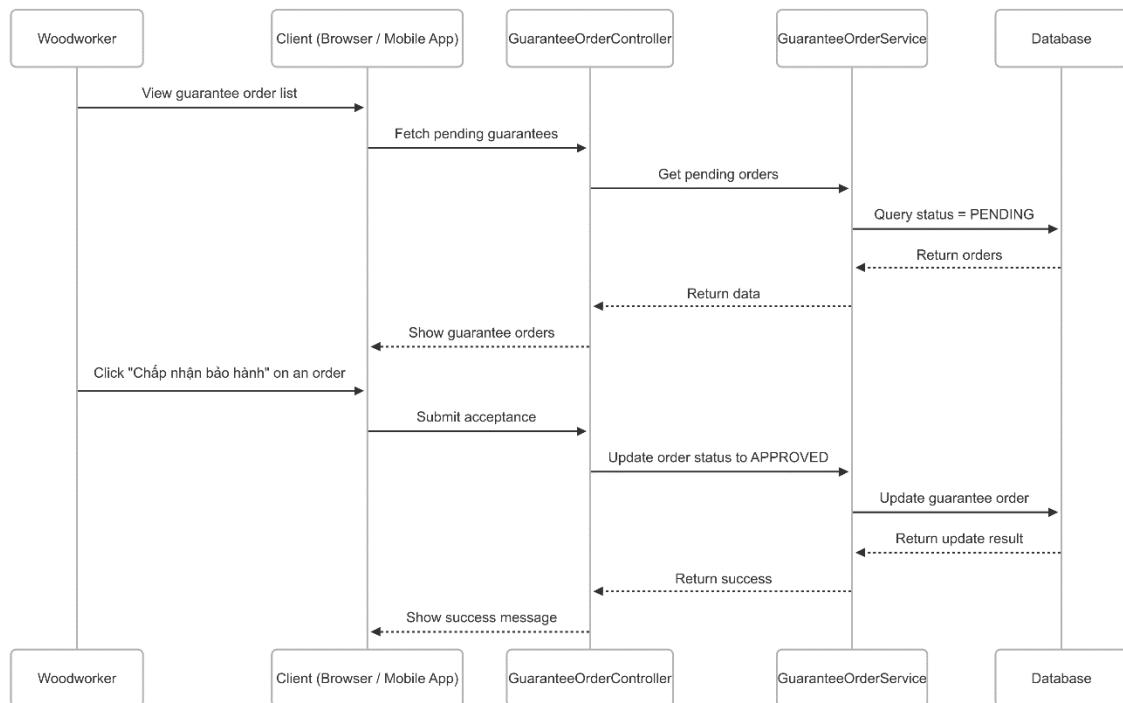


Figure 32 - Accept guarantee order sequence diagram

3.4.4 Guarantee order state machine diagram

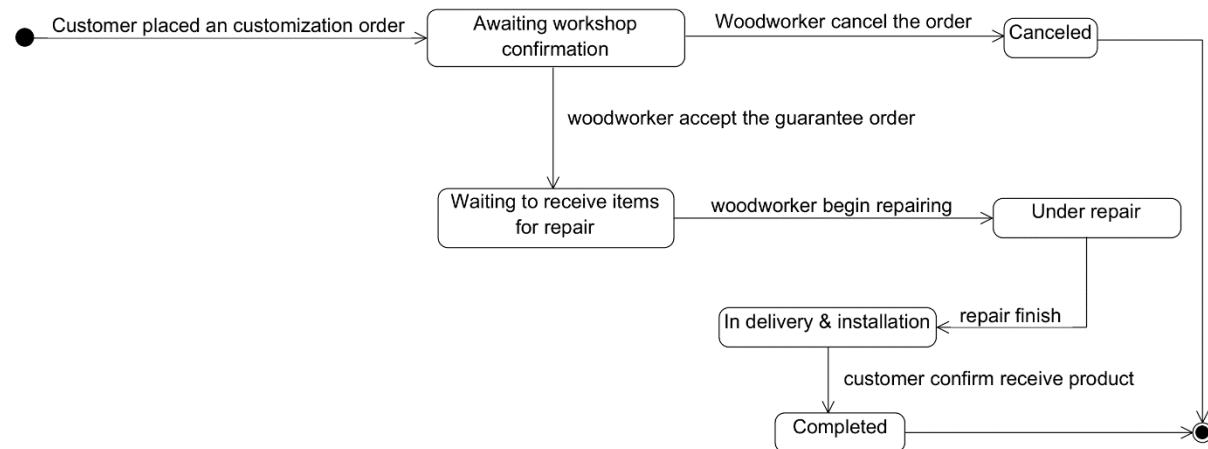


Figure 33 - Guarantee order state machine diagram

V. Software Testing Documentation

1. Overall Description

1.1 Test Scope

We plan to define test cases in main flow only.

1.2 Test Model

We use Incremental testing for some approaches:

- Each module is required a role to use clearly
- Each module could test at the early stage of the develop process

1.3 Testing Levels

- Testing Level: User Acceptance Test
- Testing Types: Functional Testing, User Interface Testing

2. Test Plan

2.1 Test Stages

Type of test	Test level			
	Unit	Integration	System	Acceptance
Functional testing				X
User interface testing				X

Table 17 - Test Stages

2.2 Resources

Worker	Role	Specific Responsibilities
Võ Trọng Luân	Leader	Planning, execute, verify test result
Trần Việt Phúc	Member	Planning, execute, verify test result
Nguyễn Đăng Phương Nam	Member	Planning, execute, verify test result
Lê Nguyên An	Member	Planning, execute, verify test result

Table 18 - Test Resources

2.3 Test Milestones

No.	Deliverables	Due date
1	Test document	30/04/2025

Table 19 - Test Milestones

3. Test Cases and Test Reports

Testing Level	File Name
User acceptance test	Test_Report.xlsx

Table 20 - Test Cases and Test Reports

VI. Release Package & User Guides

1. Deliverable Package

No.	Items	Sub-items	Type	Version
Code package				
1	Web application module	SP25SE026_GSP48_WDCRBP_WebApp.zip	New	1.0.0
2	Mobile application module	SP25SE026_GSP48_WDCRBP_Mobile.zip	New	1.0.0
3	Web API module	SP25SE026_GSP48_WDCRBP_API.zip	New	1.0.0
Database				
1	Initialise Database	db-create-script.sql; db-insert-script.sql	New	
Document				
1	Final project report	WDCRBP_Final_Report.docx		

Table 21 - Deliverable Package

2. Installation Guides

2.1 System Requirements

2.1.1 Web application

Spec	Minimum	Recommend
Internet connection	LAN, internet access capability	LAN, Wi-fi (16 Mbps)
Processor	Intel Core i5 1.5GHz	Intel Core i7 2.6GHz
Memory	4GB RAM	8GB RAM up
Storage	4GB	6GB up

Table 22 - System Requirements Web Application

2.2.2 Mobile application

Spec	Minimum	Recommend
Internet connection	3G connection, internet access capability	4G/5G Wi-Fi stable connection
Processor	Quad-core 1.5GHz	Octa-core 2.0GHz or higher
Memory (RAM)	2GB RAM	4GB RAM or more
Storage	2GB free space	4GB free space or more

Table 23 - System Requirements Mobile Application

2.2.3 Web API

Component Name	Version	Description
Operating System	Windows 10/11, Linux, Ubuntu	Run the server application
Database	MySQL 8	Data storage for application
Runtime Framework	Java 23, Spring Boot	Run backend APIs

Table 24 - System Requirements Web API

2.2 Setup Files

- db-create-script.sql; db-insert-script.sql: Database script for system
- SP25SE026_GSP48_WDCRBP_API.zip: API for services
- SP25SE026_GSP48_WDCRBP_WebApp.zip: Web application for browser use
- SP25SE026_GSP48_WDCRBP_Mobile.zip: Mobile application for mobile use

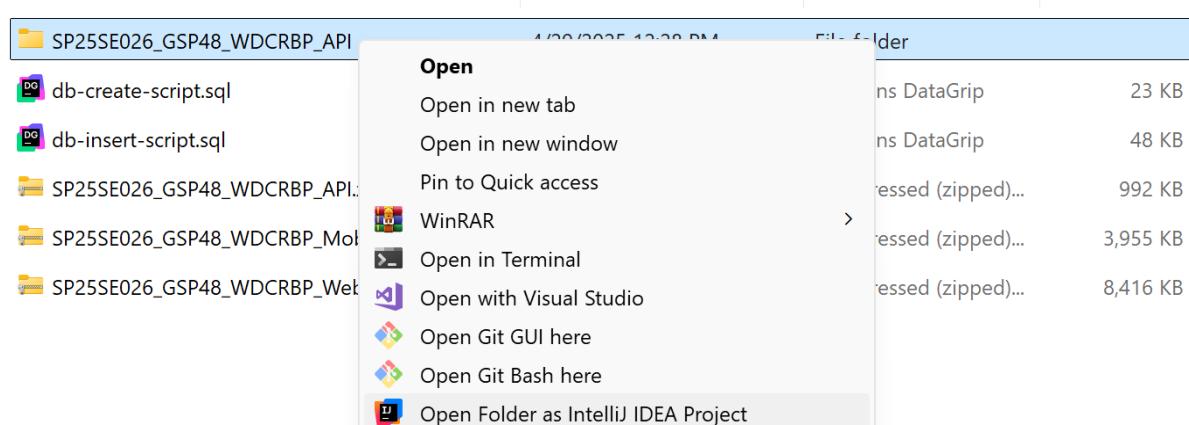
2.3 Installation Instruction

2.3.1 Setup Database

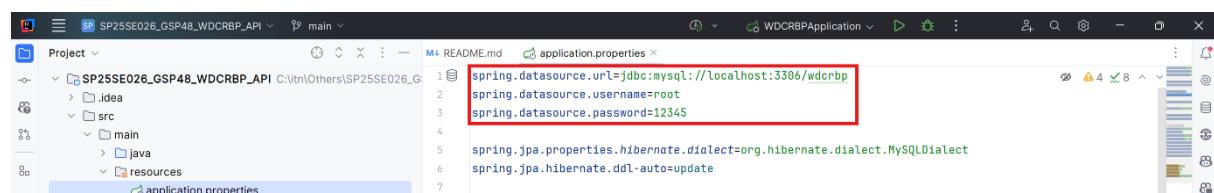
- Install MySQL 8 by following guidelines link: [MySQL download: Cài đặt MySQL trên Windows, MacOS, Linux - ITviec Blog](#)

2.3.2 Setup Back-end API

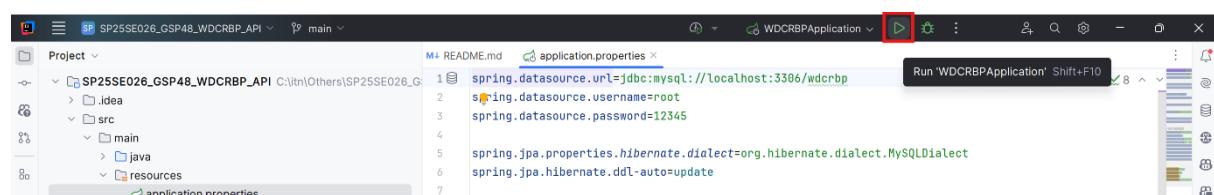
- Install IntelliJ Ultimate by following guidelines link: [Hướng dẫn cài đặt IntelliJ](#)
- Extracting: SP25SE026_GSP48_WDCRBP_API.zip
- Open folder as IntelliJ IDEA Project



- Config your database connection



- Run project



- Test project in browser by go to url: <http://localhost:8080/swagger-ui/index.html#/>

The screenshot shows the Swagger UI interface for an OpenAPI definition. At the top, it says "OpenAPI definition v0 OAS 3.0". Below that, there's a "Servers" dropdown set to "http://localhost:8080 - Generated server url". On the right, there's a green "Authorize" button. The main area is titled "woodworker-profile-controller" and contains a list of API endpoints:

- PUT** /api/v1/ww/update-status
- PUT** /api/v1/ww/update-woodworker-profile
- PUT** /api/v1/ww/update-warranty-policy
- PUT** /api/v1/ww/update-public-status
- PUT** /api/v1/ww/addServicePack
- POST** /api/v1/ww/register
- GET** /api/v1/ww
- GET** /api/v1/ww/{whId}
- GET** /api/v1/ww/user/{userId}
- GET** /api/v1/ww/inactive

2.3.3 Setup Web Application

- Install Visual Studio Code by following guidelines link: [Cài Visual Studio Code lập trình C/C++ - IT DESIGN](#)
- Install npm through Visual Studio Code or you can install npm via NodeJS [Node.js — Run JavaScript Everywhere](#)
- Extract SP25SE026_GSP48_WDCRBP_WebApp.zip and open folder with Visual Studio Code
- Open terminal and enter npm i:

The screenshot shows the Visual Studio Code terminal window. The command \$ npm i is being typed in. The output shows the user is in a directory named SP25SE026_GSP48_WDCRBP_WebApp (main).

```
trong@luan MINGW64 /c/itn/Others/SP25SE026_GSP48_WDCRBP_WebApp (main)
$ npm i
```

- Change API_URL based on your backend configuration

The screenshot shows the Visual Studio Code Explorer. The .env file is open, and the line VITE_API_URL=http://localhost:8080 is highlighted with a red box. The Explorer sidebar shows other files like .eslintrc.js, .gitignore, index.html, package-lock.json, package.json, README.md, vercel.json, and vite.config.js.

- In terminal, enter npm run dev

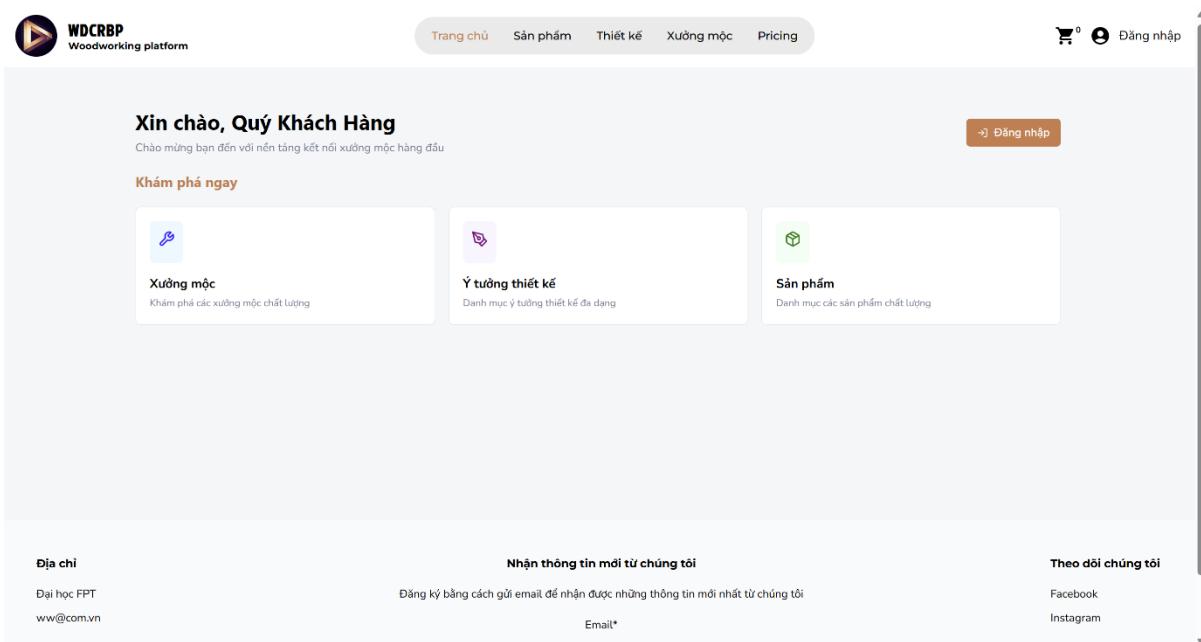
```
trong@luan MINGW64 /c/itn/Others/SP25SE026_GSP48_WDCRBP_WebApp (main)
○ $ npm run dev
```

```
> bbp-frontend@0.0.0 dev
> vite
```

VITE v4.4.11 ready in 340 ms

```
→ Local: http://localhost:5173/
→ Network: use --host to expose
→ press h to show help
```

- Project start with <http://localhost:5173/>



2.3.4 Setup Mobile Application

- Open terminal in visual studio code and enter: npm install --global yarn

```
trong@luan MINGW64 /c/itn/Others/SP25SE026_GSP48_WDCRBP_WebApp (main)
○ $ npm install --global yarn
```

- Install Bluestack by following guidelines link: [Cách cài đặt giả lập BlueStacks để chơi game Android trên máy tính, PC](#)
- Run Bluestack and install Expo Go: [Expo Go - Ứng dụng trên Google Play](#)
- Extract SP25SE026_GSP48_WDCRBP_Mobile.zip and open folder with Visual Studio Code
- Open terminal and enter yarn install:

```
trong@luan MINGW64 /c/itn/Others/SP25SE026_GSP48_WDCRBP_Mobile (main)
○ $ yarn install
```

- Change API_URL based on your backend configuration

```

File Edit Selection View Go Run ...
SP25SE026_GSP48_WDCRBP_Mobile
EXPLORER
SP25SE026_GSP48_WDCRBP_Mobile
src > config > index.js > ...
1 const publicRuntimeConfig = {
2   API_URL: "http://10.0.2.2:8080",
3 }
4
5 export const { API_URL, APP_URL } = publicRuntimeConfig;
6
7 export default publicRuntimeConfig;
8

```

- In terminal enter yarn start

```

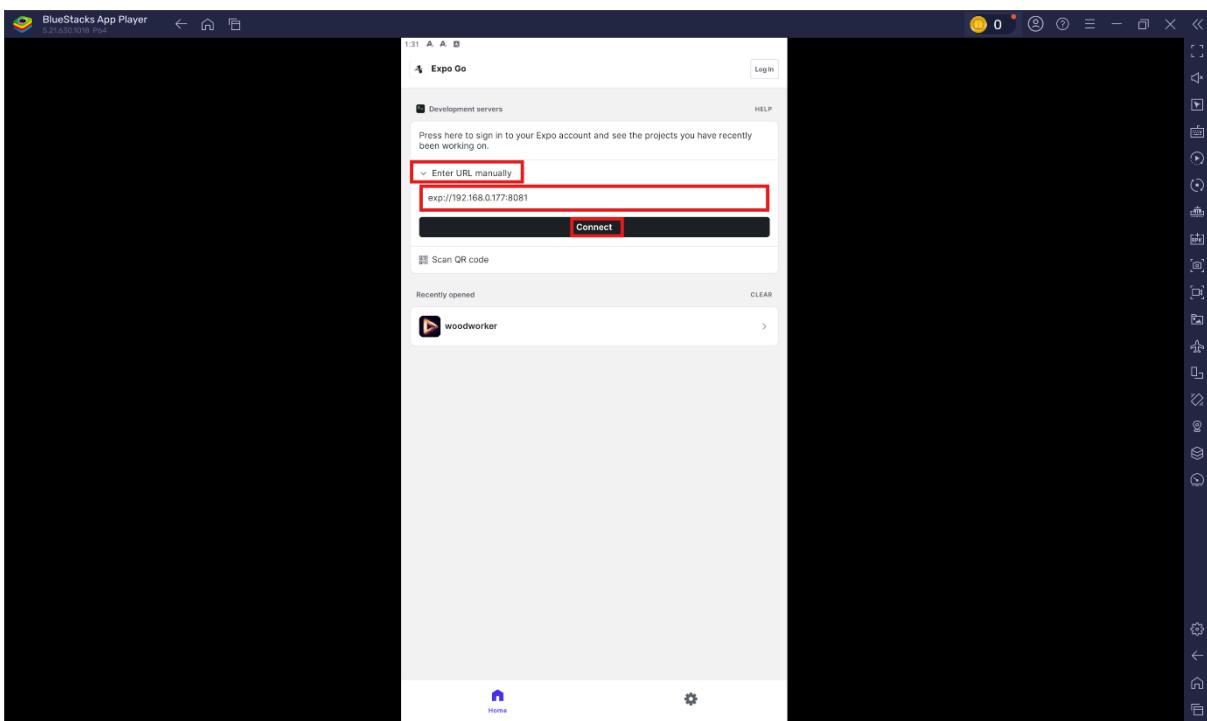
trong@luan MINGW64 /c/itn/Others/SP25SE026_GSP48_WDCRBP_Mobile (main)
$ yarn start
yarn run v1.22.22
$ expo start
Starting project at C:/itn/Others/SP25SE026_GSP48_WDCRBP_Mobile
Starting Metro Bundler
The following packages should be updated for best compatibility with the installed expo version:
  @react-native-async-storage/async-storage@2.1.2 - expected version: 1.23.1
  expo@52.8.42 - expected version: ~52.0.46
  react-native@0.76.8 - expected version: 0.76.9
Your project may not work correctly until you install the expected versions of the packages.

QR code
  ▶ Metro waiting on exp://192.168.0.177:8081
  Scan the QR code above with Expo Go (Android) or the Camera app (iOS)

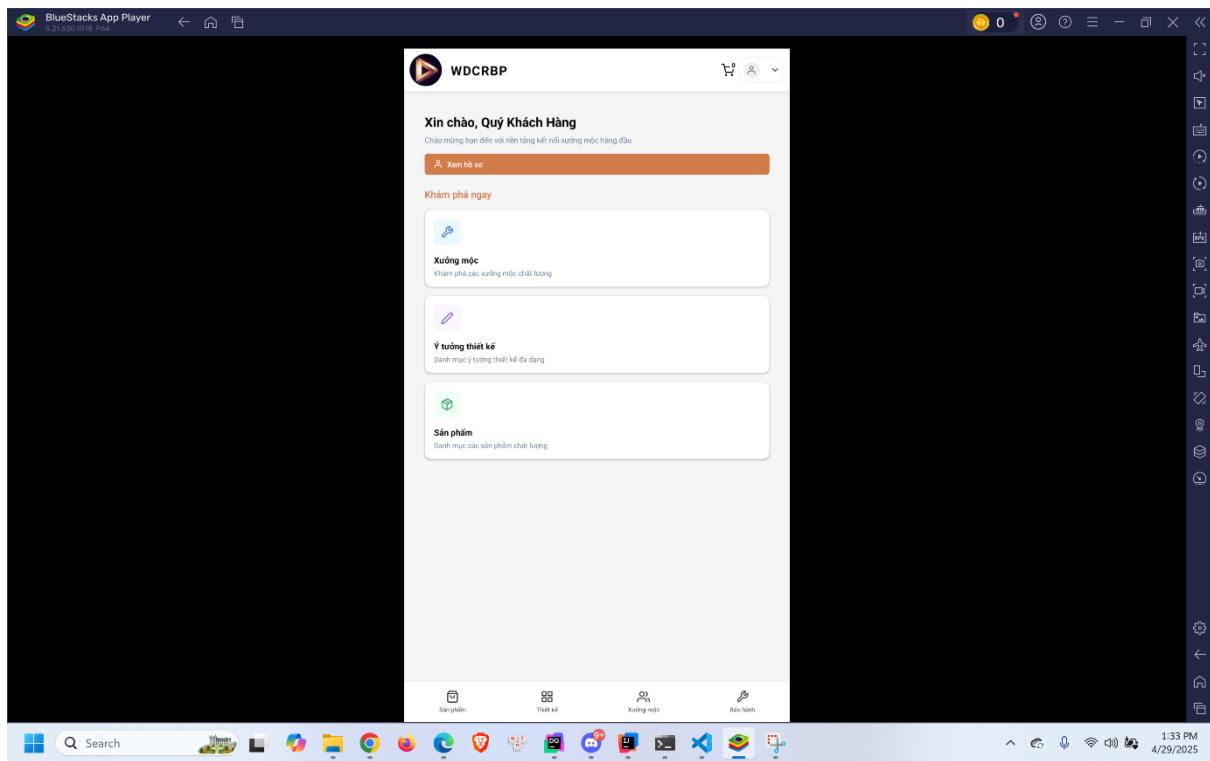
  ▶ Using Expo Go

```

- Open Expo Go in BlueStack and enter URL: exp://192.168.0.177:8081 then click Connect



The application will start after connecting



3. User Manual

3.1 Overview

Our platform is a dynamic marketplace connecting woodworkers and customers for personalized furniture solutions. Users can register, offer, or book services seamlessly, focusing on customization, personalization, and guarantee services to ensure high-quality, made-to-order products. The system is designed to empower woodworkers to showcase their craftsmanship and enable customers to easily configure and request.

3.2 Web application

3.2.1 Woodworker Registration Feature

3.2.1.1 Guest register as a woodworker

Xin chào, Quý Khách Hàng
Chào mừng bạn đến với nền tảng kết nối xưởng mộc hàng đầu

Khám phá ngay

Xưởng mộc
Khám phá các xưởng mộc chất lượng

Ý tưởng thiết kế
Danh mục ý tưởng thiết kế đa dạng

Sản phẩm
Danh mục các sản phẩm chất lượng

Địa chỉ
Đại học FPT
vv@com.vn

Nhận thông tin mới từ chúng tôi
Đăng ký bằng cách gửi email để nhận được những thông tin mới nhất từ chúng tôi

Email*

Theo dõi chúng tôi
Facebook
Instagram

Đăng nhập

Email / Số điện thoại *

Nhập email hoặc số điện thoại

Mật khẩu *

Nhập mật khẩu

Đăng nhập

Phương thức đăng nhập

Mật khẩu Email OTP

Quên mật khẩu Đăng ký

Đăng ký trở thành xưởng mộc

Địa chỉ
Đại học FPT
vv@com.vn

Nhận thông tin mới từ chúng tôi
Đăng ký bằng cách gửi email để nhận được những thông tin mới nhất từ chúng tôi

Email*

Theo dõi chúng tôi
Facebook
Instagram

Đăng ký thông tin xưởng mộc

Thông tin người đại diện

Họ và tên * Email * Số điện thoại *

Nhập họ và tên Nhập email Nhập số điện thoại

Thông tin xưởng mộc

Tên xưởng mộc * Loại hình kinh doanh *

Nhập tên xưởng mộc Chọn loại hình

Địa chỉ Tỉnh/Thành phố * Số nhà, tên đường *

Nhập tên đường, số nhà Chọn tỉnh/thành phố

Quận/Huyện * Chọn quận/huyện

Phường/Xã * Chọn phường/xã

Mã số thuế * Nhập mã số thuế

Giới thiệu *

Mộc Hưng Phát là xưởng mộc thủ công "cực chất" chuyên chế tác nội thất và décor hoàn toàn từ gỗ tự nhiên. Với tinh huyết giữ trọn hồn mộc mạc, mỗi món đồ nơi đây không chỉ là sản phẩm mà còn là dấu ấn sáng tạo - vừa tinh tế vừa bền bỉ. Chúng minh nói không với nhựa hay kim loại, cam kết mang đến bạn góc sống gần gũi với thiên nhiên, toát lên phong cách riêng biệt của giá trị.

Ảnh đại diện cho xưởng *

Nhập ảnh

Xác nhận thông tin

Tôi đã kiểm tra thông tin và xác nhận thoả tác

Gửi **Đăng ký**

Dịa chỉ **Nhận thông tin mới từ chúng tôi** **Theo dõi chúng tôi**

Đại học FPT
www.com.vn

Email*

Gửi ngay

Nhận hỗ trợ

Facebook
Instagram

- Click "Đăng nhập"
- Click "Đăng ký" trở thành xưởng mộc"
- Fill in the registration form
- Click "Đăng ký"

3.2.1.2 Admin approve woodworker registration

WDCRP Woodworking platform

Menu quản trị viên

Tổng quan

Đơn đăng ký xưởng

Giao dịch

Cấu hình ti lệ cọc dịch vụ

Gói dịch vụ

Khiếu nại

Người dùng

Cấu hình hệ thống

Hỗ trợ

Chi tiết đăng ký xưởng mộc

Ảnh đại diện

Thông tin người đại diện

Mã xưởng mộc: 4

Họ và tên: Nguyễn Văn Hùng

Email: trongluan103@gmail.com

Số điện thoại: 0912345678

Thông tin xưởng mộc

Tên xưởng mộc: Xưởng Mộc Hưng Phát

Loại hình kinh doanh: Cá nhân

Mã số thuế: 6001790150

Địa chỉ: 12A Nguyễn Đình Chiểu, Phường Tân Định, Quận 1, Hồ Chí Minh

Giới thiệu:

Mộc Hưng Phát là xưởng mộc thủ công "cực chất" chuyên chế tác nội thất và décor hoàn toàn từ gỗ tự nhiên. Với tâm huyết giữ trọn hồn mộc mạc, mỗi món đồ nơi đây không chỉ là sản phẩm mà còn là dấu ấn sáng tạo – vừa tinh tế vừa bền bỉ. Chúng minh nói không với nhựa hay kim loại, cam kết mang đến cho bạn góc sống gần gũi với thiên nhiên, toát lên phong cách riêng biệt của già chú.

Ghi chú

Email: trongluan103@gmail.com

Số điện thoại: 0912345678

Tên xưởng mộc: Xưởng Mộc Hưng Phát

Loại hình kinh doanh: Cá nhân

Mã số thuế: 6001790150

Địa chỉ: 12A Nguyễn Đình Chiểu, Phường Tân Định, Quận 1, Hồ Chí Minh

Giới thiệu:

Mộc Hưng Phát là xưởng mộc thủ công "cực chất" chuyên chế tác nội thất và décor hoàn toàn từ gỗ tự nhiên. Với tâm huyết giữ trọn hồn mộc mạc, mỗi món đồ nơi đây không chỉ là sản phẩm mà còn là dấu ấn sáng tạo – vừa tinh tế vừa bền bỉ. Chúng minh nói không với nhựa hay kim loại, cam kết mang đến cho bạn góc sống gần gũi với thiên nhiên, toát lên phong cách riêng biệt của già chú.

Ghi chú

Nhập ghi chú về việc duyệt/từ chối đăng ký

Xác nhận thông tin

Tôi đã kiểm tra thông tin và xác nhận thao tác

x Đóng Tự chối Duyệt

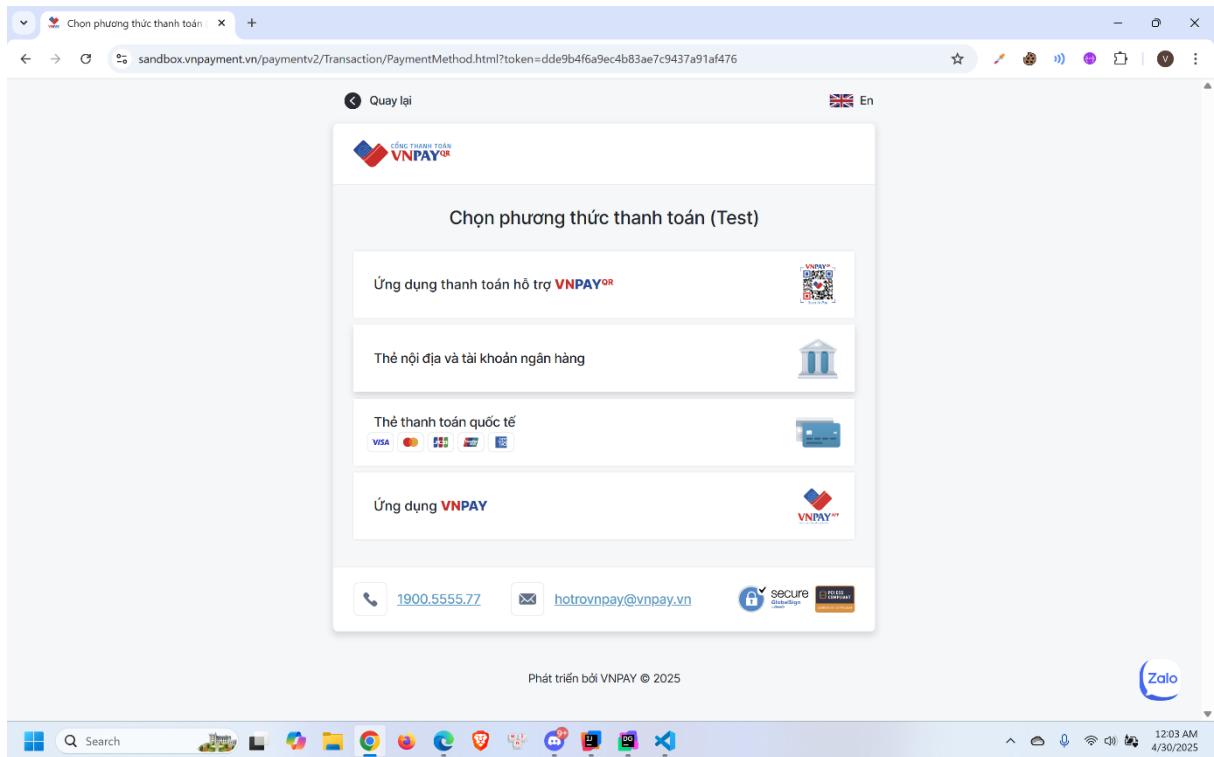
Trang dành cho quản trị viên nền tảng WDCRP

- Click “Đơn đăng ký xưởng”
- Click “Chi tiết”
- Click “Duyệt”

3.2.1.3 Woodworker deposit money to wallet

The screenshot shows the 'Quản lý Ví' (Wallet Management) section of the WDCRPB platform. At the top, there's a green button labeled '+ Nạp tiền' (Deposit Money) and a purple button labeled '- Rút tiền' (Withdraw Money). Below this, a table titled 'Các khoản giao dịch' (Transaction History) is displayed, showing the message 'No Rows To Show'. The top navigation bar includes links for Trang chủ (Home), Sản phẩm (Products), Thiết kế (Design), and Xưởng mộc (Woodshop). On the right, there are buttons for 'Công khai xưởng' (Public Workshop) and 'Xem xưởng của bạn' (View your workshop).

This screenshot shows the same wallet management interface as above, but with a modal dialog box titled 'Nạp tiền' (Deposit) overlaid. The dialog contains a text input field labeled 'Nhập số tiền cần nạp' (Enter the amount to deposit) and two buttons: 'Đóng' (Close) and 'Nạp tiền' (Deposit). The background page remains largely the same, with the transaction history table showing 'No Rows To Show'.



1. Click “Nạp tiền”
2. Enter amount and click “Nạp tiền”

3.2.1.4 Woodworker upgrade service pack

The screenshot displays two browser windows of the WDCRP platform.

Top Window (Screenshot 1):

- Header:** WDCRP, Demo Cổng thanh toán VNPAY.
- Navigation:** Trang chủ, Sản phẩm, Thiết kế, Xưởng mộc.
- Right Side:** Công khai xưởng, Xem xưởng của bạn.
- Content:**
 - Menu xưởng mộc:** Ví, Hồ sơ.
 - Quản lý Gói dịch vụ:**
 - Thông tin gói dịch vụ:** Ngày bắt đầu: Chưa đăng ký, Ngày kết thúc: Chưa đăng ký.
 - Buttons:** + Mua bằng ví, + Mua qua cổng thanh toán.
 - Note:** Xem hướng dẫn cách quy đổi ngày khi nâng cấp gói dịch vụ.
 - Quản lý Thông tin cá nhân:**
 - Thông tin người đại diện:** Họ và tên * (Nguyễn Văn Hùng), Email * (trongluan103@gmail.com), Số điện thoại * (0912345678).
 - Xác nhận thông tin:** Tôi đã kiểm tra thông tin và xác nhận thao tác. Cập nhật thông tin.
 - Đổi mật khẩu:** Mật khẩu hiện tại *, Mật khẩu mới *.

Bottom Window (Screenshot 2):

- Header:** WDCRP, Demo Cổng thanh toán VNPAY.
- Navigation:** Trang chủ, Sản phẩm, Thiết kế, Xưởng mộc.
- Right Side:** Công khai xưởng, Xem xưởng của bạn.
- Content:**
 - Mua gói dịch vụ:**
 - Các gói dịch vụ dành cho xưởng mộc:** Tham gia cùng hàng nghìn Xưởng mộc dịch vụ khác.
 - Hàng tháng, Hàng quý, Hàng năm:**
 - Gói Đồng:** 200,000 đồng/tháng. Kích hoạt: Quản lý dịch vụ cung cấp (Tùy chỉnh, sửa chữa), Quản lý tương thiết kế, Quản lý đơn hàng dịch vụ, Trang cá nhân (Profile) (Giới thiệu, thông tin, hình ảnh), 5 bài đăng trên trang cá nhân/tháng. Chi tiết: Quản lý sản phẩm & bán sản phẩm có sẵn, Ưu tiên hiển thị trong kết quả tìm kiếm, Chức năng cung cấp dịch vụ cá nhân hóa.
 - Gói Bạc:** 350,000 đồng/tháng. Kích hoạt: Quản lý dịch vụ cung cấp (Tùy chỉnh, sửa chữa), Quản lý tương thiết kế, Quản lý đơn hàng dịch vụ, Trang cá nhân (Profile) (Giới thiệu, thông tin, hình ảnh), 10 bài đăng trên trang cá nhân/tháng. Chi tiết: Quản lý sản phẩm & bán sản phẩm có sẵn, Ưu tiên hiển thị trong kết quả tìm kiếm, Chức năng cung cấp dịch vụ cá nhân hóa.
 - Gói Vàng:** 500,000 đồng/tháng. Kích hoạt: Quản lý dịch vụ cung cấp (Tùy chỉnh, sửa chữa), Quản lý tương thiết kế, Quản lý đơn hàng dịch vụ, Trang cá nhân (Profile) (Giới thiệu, thông tin, hình ảnh), 20 bài đăng trên trang cá nhân/tháng. Chi tiết: Quản lý sản phẩm & bán sản phẩm có sẵn, Ưu tiên cao nhất trong kết quả tìm kiếm, Chức năng cung cấp dịch vụ cá nhân hóa.

- Click “Mua bằng ví” or “Mua qua cổng thanh toán”
- Choose pack to use and click “Kích hoạt”

3.2.2 Customization Service Booking Feature

3.2.2.1 Customer request a customization order

WDCRPB Demo Cổng thanh toán VNPay

Danh sách thiết kế

Tìm thấy 3 kết quả

Image	Name	Location	Rating
	Giường Gỗ Mẫu Hoàng Gia Ho...	Mộc Châm 123 Nguyễn Văn Linh, Phường Tăng Nhơn Phú A, Thành Phố Thủ Đức, Hồ Chí Minh	★★★★★
	Tủ quần áo hiện đại	Bình Minh 23 đường Trần Thái, Phường Phước Long A, Thành Phố Thủ Đức, Hồ Chí Minh	★★★★★
	Bàn trang điểm Á Châu mẫu...	Hoa Nam 12/12 đường Đồng Khởi, Xã Hòa Khương, Huyện Hòa Vang, Đà Nẵng	★★★★★

Bộ lọc

- Danh mục
- Chọn danh mục
- Tỉnh thành
- Loại xưởng
- Lọc theo số sao
- Sắp xếp theo
- Tên thiết kế
- Áp dụng bộ lọc
- Áp dụng từ khóa

Lọc

Địa chỉ

Đại học FPT
www.fpt.edu.vn
https://wdcrbp.vercel.app/design/

Nhận thông tin mới từ chúng tôi

Đăng ký bằng cách gửi email để nhận được những thông tin mới nhất từ chúng tôi

Email*

Theo dõi chúng tôi

Facebook
Instagram

12:07 AM 4/30/2025

Chi tiết thiết kế

Giường Gỗ Mẫu Hoàng Gia Hoa Lá Tây

Loại sản phẩm: Giường truyền thống

Lắp đặt: Không cần lắp đặt

Mô tả:

ĐẶC ĐIỂM NỔI BẬT CỦA SẢN PHẨM

- ĐƯỢC ĐỨC CHẠM BẮNG TAY bởi các nghệ nhân làng nghề gỗ ĐỒNG KÝ BẮC NINH
- KẾT CẤU SẢN PHẨM vô cùng vững chắc được kết nối khớp móng
- GIÁ CÁ CANH TRATHANH bởi được sản xuất tại xưởng làng nghề gỗ ĐỒNG KÝ đã nổi tiếng từ rất lâu về truyền thống làng nghề về gỗ.
- CÓ CÁCH NHÀNG 2 MIỀN BẮC - NAM để dễ dàng giao nhận hàng hóa giúp quý khách hàng dễ dàng mua hàng và đến tham quan mua sắm
- NHÂN GIA CÔNG SẢN XUẤT THEO MÀU

Cấu hình sản phẩm

Kích thước (dài x rộng x cao - cm)

100 x 100 x 100 90 x 90 x 90

Loại gỗ

Xoan Sồi

5.000.000 đ

ĐẶT NGAY **Thêm vào giỏ**

12:08 AM 4/30/2025

The screenshot displays two browser windows of the WDCRP platform.

Top Window (Product Detail Page):

- Header:** WDCRP, Demo Cổng thanh toán VNPAY.
- Page Title:** Chi tiết thiết kế.
- Content:**
 - Product Image:** Giường Gỗ Mẫu Hoàng Gia Hoa Lá Tây.
 - Product Description:** Giường truyền thống.
 - Delivery Information:** Không cần lắp đặt.
 - Product Details:** Kích thước (dài x rộng x cao - cm): 100 x 100 x 100; Loại gỗ: Xoan.
 - Price:** 5.000.000 đ.
 - Buttons:** ĐẶT NGAY, Thêm vào giỏ.
- Right Panel:** Giò hàng (Cart) showing the same item with the same details and quantity SL: 1.

Bottom Window (Shopping Cart Page):

- Header:** WDCRP, wdcrbp.vercel.app/cart?tab=design.
- Content:**
 - Cart Item:** Xưởng mộc: Mộc Chạm, Không cần lắp đặt, Giường Gỗ Mẫu Hoàng Gia Hoa Lá Tây, Kích thước (dài x rộng x cao - cm): 100 x 100 x 100, Loại gỗ: Xoan, Giá: 5.000.000 đ.
 - Quantity:** Số lượng: 1.
 - Right Panel:** Thông tin đặt hàng (Order Information) including:
 - Đơn hàng không cần lắp đặt.
 - Thông tin xưởng mộc: Mộc Chạm, 123 Nguyễn Văn Linh, Phường Tăng Nhơn Phú A, Thành Phố Thủ Đức, Hồ Chí Minh.
 - Địa chỉ giao hàng: 123 Nguyễn Văn Linh, Phường Hòa Minh, Quận Liên Chiểu, Đà Nẵng.
 - Ghi chú đơn hàng: Giao hành nhanh giúp em.
 - Tiền sản phẩm: 5.000.000 đ.
 - Phi vận chuyển: 605.000 đ.
 - Tổng giá trị: 5.605.000 đ.
 - Buttons:** Tiến hành đặt hàng.

- Click “Thiết kế” in the header
- Select design to view detail
- Click “Thêm vào giỏ”
- Click cart icon in the header
- Click “Xem giò hàng”
- Fill in the form and click “Tiến hành đặt hàng”

3.2.2.2 Woodworker book a consultant appointment

Chi tiết đơn #1 Đang chờ xưởng xác nhận đơn hàng

Thông tin sản phẩm

#1. Giường Gỗ Mẫu Hoàng Gia Họa Lá Tây x 1
GIƯỜNG TRUYỀN THỐNG

Phi vận chuyển: 605.000đ

Thành tiền: 5.605.000đ

Thông tin đơn hàng

Mã đơn hàng: 1
Loại dịch vụ: Tùy chỉnh
Ngày đặt: 30/04/2025 00:27
Số lượng sản phẩm: 1
Yêu cầu lắp đặt bởi xưởng: Không cần lắp đặt
Ghi chú:

Thông tin lịch hẹn tư vấn bàn hợp đồng

Không có lịch hẹn tư vấn

Cập nhật lịch hẹn

Thông tin lịch hẹn

Hình thức: Chọn hình thức
Địa điểm: Địa điểm
Ngày hẹn: mm/dd/yyyy --::--
Mô tả: Mô tả

Xác nhận thông tin

Tôi đã kiểm tra thông tin và xác nhận thảo tác

Cập nhật

Thống kê

Mã đơn hàng: 1
Loại dịch vụ: Tùy chỉnh
Ngày đặt: 30/04/2025 00:27
Số lượng sản phẩm: 1
Yêu cầu lắp đặt bởi xưởng: Không cần lắp đặt
Ghi chú:

Xưởng mộc Mộc Chạm

- Click “Cập nhật lịch hẹn tư vấn về hợp đồng”
- Enter the form and click “Cập nhật”

3.2.2.3 Customer confirm consultant appointment

The screenshot shows the WDCRBP Woodworking platform's order detail page. The main content includes:

- Thông tin sản phẩm**: #1. Giường Gỗ Mẫu Hoàng Gia Họa Lá Tây x 1 (GIƯỜNG TRUYỀN THỐNG)
- Phi vận chuyển**: 605.000 đ
- Thành tiền**: 5.605.000 đ
- Thông tin đơn hàng**:
 - Mã đơn hàng: 1
 - Loại dịch vụ: Tùy chỉnh
 - Ngày đặt: 30/04/2025 00:27
 - Số lượng sản phẩm: 1
 - Yêu cầu lắp đặt bởi xưởng: Không cần lắp đặt
 - Ghi chú: Giao hành nhanh giúp em
- Thông tin lịch hẹn tư vấn bàn hợp đồng**:
 - Hình thức: Online
 - Địa điểm: Google meeting https://meet.google.com/dsj3sjw
 - Ngày giờ hẹn: lúc 13:30 1 tháng 5, 2025
 - Mô tả: Bản vẽ hợp đồng

A modal window titled "Xác nhận lịch hẹn" (Confirm Appointment) is displayed, containing the following information:

- Chi tiết lịch hẹn**:
 - Ngày hẹn: lúc 13:30 1 tháng 5, 2025
 - Hình thức: Online
 - Địa điểm: Google meeting https://meet.google.com/dsj3sjw
 - Mô tả: Bản vẽ hợp đồng
- Xác nhận thông tin**:
 - Tôi đã kiểm tra thông tin và xác nhận thoả tác

Buttons at the bottom of the modal include "Đóng" (Close) and "Xác nhận" (Confirm).

- Click “Xác nhận lịch hẹn”

- Click “Xác nhận”

3.2.2.4 Woodworker update the contract

The screenshot displays two consecutive pages from the WDCRPB Woodworking platform.

Page 1: Chi tiết đơn #1

- Left sidebar:** Menu xưởng mộc (Workshop menu) with items: Đơn hàng, BH & Sửa chữa, Dịch vụ, Thiết kế, Sản phẩm, Bài đăng, Khiếu nại, Đánh giá, Ví, Hỗ trợ.
- Header:** WDCRPB Woodworking platform, Trang chủ, Sản phẩm, Thiết kế, Xưởng mộc, Công khai xưởng, Xem xưởng của bạn.
- Content:**
 - Chi tiết đơn #1**: Đã duyệt lịch hẹn.
 - Chung**, **Tiền độ**, **Hợp đồng & Giao dịch**.
 - Thông tin sản phẩm**: #1. Giường Gỗ Mẫu Hoàng Gia Hoa Lá Tây x 1 (GIƯỜNG TRUYỀN THỐNG).
 - Phi vận chuyển:** 605.000 ₫
 - Thành tiền:** 5.605.000 ₫
 - Thông tin đơn hàng**: Mã đơn hàng: 1, Loại dịch vụ: Tùy chỉnh, Ngày đặt: 30/04/2025 00:27, Số lượng sản phẩm: 1, Yêu cầu lắp đặt bởi xưởng: Không cần lắp đặt, Ghị chú:.
 - Thông tin lịch hẹn tư vấn bàn hợp đồng**: Hình thức: Online, Địa điểm: Google meeting https://meet.google.com/dsj3sjw, Ngày giờ hẹn: lúc 13:30 1 tháng 5, 2025, Mô tả: Bàn về hợp đồng.

Page 2: Thông tin hợp đồng

- Left sidebar:** Menu xưởng mộc (Workshop menu) with items: Đơn hàng, BH & Sửa chữa, Dịch vụ, Thiết kế, Sản phẩm, Bài đăng, Khiếu nại, Đánh giá, Ví, Hỗ trợ.
- Header:** WDCRPB Woodworking platform, Công khai xưởng, Xem xưởng của bạn.
- Content:**
 - Điều khoản của nền tảng:** Xem điều khoản nền tảng.
 - Điều khoản của xưởng mộc:** * Yêu cầu thanh toán đúng hạn và kịp thời cung cấp thông tin khi cần thiết.
 - Thời hạn bảo hành theo sản phẩm (tháng):** * MÃ SẢN PHẨM: 1, SẢN PHẨM: Giường truyền thống, SỐ LƯỢNG: 1, THỜI HẠN BẢO HÀNH (THÁNG): 12.
 - Ngày hoàn thành:** * 05/30/2025.
 - Chữ ký thư**: Chiều rộng (px): 500, Chiều cao (px): 200. Vui lòng ký tên vào khu vực bên dưới.

- Click “Soạn hợp đồng”
- Enter the form and sign in the box
- Click “Cập nhật”

3.2.2.5 Customer confirm contract and process payment

WDCRBP

https://wdcrbp.vercel.app/cus/service-order/1

Trang chủ Sản phẩm Thiết kế Xưởng mộc Sửa chữa / bảo hành

Gửi phản hồi Xác nhận hợp đồng

Menu khách hàng

Trang chủ Đơn hàng BH & Sửa chữa Ví Khiếu nại Hồ sơ

Chi tiết đơn #1 Đang chờ khách hàng duyệt hợp đồng

Chung Tiền độ Hợp đồng & Giao dịch

Thông tin sản phẩm

#1. Giường Gỗ Mẫu Hoàng Gia Họa Lá Tây x 1
GIƯỜNG TRUYỀN THỐNG

5.000.000 đ

Phí vận chuyển: 605.000 đ

Thành tiền: 5.605.000 đ

Search 1:12 AM 4/30/2025

WDCRBP https://wdcrbp.vercel.app/cus/service-order/1

Trang chủ Đơn hàng BH & Sửa chữa Ví Khiếu nại Hồ sơ

Tổng tiền thanh toán: 5.605.000 VND

Thị thực hiện: Thợ Test Nền Tảng

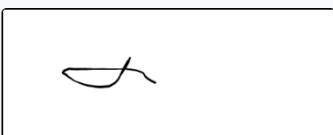
Chữ ký thợ:

Ký tên xác nhận hợp đồng

Chữ ký của bạn

Chiều rộng (px): 500 Chiều cao (px): 200

Vui lòng ký tên vào khu vực bên dưới



Lưu Xóa

Xác nhận thông tin

Tôi đã kiểm tra thông tin và xác nhận thoả tác

Đóng Xác nhận hợp đồng

Giao hành nhanh giúp em

The screenshot shows the WDCRBP Woodworking platform's service order detail page. The main content includes:

- Thông tin sản phẩm**: #1. Giường Gỗ Mẫu Hoàng Gia Họa Lá Tây x 1 (GIƯỜNG TRUYỀN THỐNG)
- Phí vận chuyển**: 605.000 đ
- Thành tiền**: 5.605.000 đ
- Thông tin đơn hàng**:
 - Mã đơn hàng: 1
 - Loại dịch vụ: Tùy chỉnh
 - Ngày đặt: 30/04/2025 00:27
 - Số lượng sản phẩm: 1
 - Yêu cầu lắp đặt bởi xưởng: Không cần lắp đặt
 - Ghi chú: Giao hành nhanh giúp em
- Thông tin lịch hẹn tư vấn bàn hợp đồng**:
 - Hình thức: Online
 - Địa điểm: Google meeting https://meet.google.com/dsj3sjw
 - Ngày giờ hẹn: lúc 13:30 1 tháng 5, 2025
 - Mô tả: Bàn về hợp đồng

A modal window titled "Thanh toán đặt cọc" (Payment for deposit) is displayed, containing the following fields:

- Mã đơn dịch vụ: #1
- Đặt cọc lần: 1
- Phản trả: 60%
- Số tiền: 3.363.000 đ
- Chọn phương thức thanh toán:
 - Thanh toán bằng ví
 - Thanh toán qua cổng thanh toán
- Xác nhận thông tin:
 - Tôi đã kiểm tra thông tin và xác nhận thao tác

The modal also includes buttons for "Đóng" (Close) and "Thanh toán" (Process payment).

- Click “Xác nhận hợp đồng”
- Check information and sign in the box
- Click “Xác nhận hợp đồng”
- Click “Thanh toán”
- Choose payment method and process payment

3.2.2.6 Woodworker upload the finish product image

The screenshot shows a web browser displaying the WDCRPB Woodworking platform. The URL is wdcrbp.vercel.app/ww/service-order/1. The main content area shows a service order detail page for a customer named "Đang giao công". The order includes a single item: "Giường Gỗ Mẫu Hoàng Gia Họa Lá Tây x 1" (Bed Frame Model Royal Lacquer Western Style x 1) at a price of 5,000,000đ. Shipping cost is 605,000đ, and the total amount is 5,605,000đ. The payment status is "Đã thanh toán" (Paid). On the left sidebar, there is a navigation menu with categories like Đơn hàng, BH & Sửa chữa, Dịch vụ, Thiết kế, Sản phẩm, Bài đăng, Khiếu nại, Đánh giá, Ví, and Hồ sơ.

Chi tiết đơn #1 (Đang giao công)

Thông tin sản phẩm

#1. Giường Gỗ Mẫu Hoàng Gia Họa Lá Tây x 1
GIƯỜNG TRUYỀN THỐNG

5.000.000đ

Phi vận chuyển: 605.000đ

Thành tiền: 5.605.000đ

Thông tin đơn hàng

Mã đơn hàng: 1
Loại dịch vụ: Tùy chỉnh
Ngày đặt: 30/04/2025 08:00
Số lượng sản phẩm: 1
Yêu cầu lắp đặt bởi xưởng: Không cần lắp đặt
Ghi chú:

Thông tin lịch hẹn tư vấn bàn hợp đồng

Hình thức: Online
Địa điểm: Google meeting <https://meet.google.com/azdhuw>
Ngày giờ hẹn: lúc 12:00 30 tháng 4, 2025
Mô tả: Họp bàn hợp đồng

Cập nhật ảnh hoàn thiện sản phẩm và giao hàng

Đã chuẩn bị 0/1 sản phẩm

Sản phẩm #1 - Giường truyền thống

Tải lên ảnh hoàn thiện mới:

Kéo thả ảnh vào đây hoặc click để chọn (Tối đa 5 ảnh)
Hỗ trợ: JPG, JPEG, PNG, GIF (Tối đa 5MB)

Hiện ảnh ban đầu

Đóng

5.000.000đ

605.000đ

5.605.000đ

Xưởng mộc Mộc Chạm

- Click “Cập nhật ảnh hoàn thiện và giao hàng”
- Upload images and click “Cập nhật”

3.2.2.7 Customer confirm receive product and process payment

The screenshot shows a web browser window for the WDCRP platform. The URL is wdcrbp.vercel.app/cus/service-order/1. The page displays a service order detail (#1) for a "Giường Gỗ Mẫu Hoàng Gia Họa Lá Tây x 1". The total amount is 5.605.000đ. The left sidebar shows navigation options like Trang chủ, Đơn hàng, BH & Sửa chữa, Ví, Khiếu nại, and Hồ sơ.

Chi tiết đơn #1 (Đang giao hàng lắp đặt)

Thông tin sản phẩm

#1. Giường Gỗ Mẫu Hoàng Gia Họa Lá Tây x 1
GIƯỜNG TRUYỀN THỐNG

Phi vận chuyển: 605.000đ

Thành tiền: 5.605.000đ

Thông tin đơn hàng

Mã đơn hàng: 1
Loại dịch vụ: Tùy chỉnh
Ngày đặt: 30/04/2025 08:00
Số lượng sản phẩm: 1
Yêu cầu lắp đặt bởi xưởng: Không cần lắp đặt
Ghi chú: Giao hàng nhanh giúp em

Thông tin lịch hẹn tư vấn bàn hợp đồng

Hình thức: Online
Địa điểm: Google meeting <https://meet.google.com/azdhuw>
Ngày giờ hẹn: lúc 12:00 30 tháng 4, 2025
Mô tả: Họp bàn hợp đồng

The screenshot shows a payment modal titled "Thanh toán đặt cọc" (Partial payment) overlaid on the service order detail page. The modal contains fields for the number of items (2), payment percentage (40%), and the amount (2.242.000đ). It also includes a payment method selection (Bank card) and a checkbox for confirming information.

Chi tiết đơn #1

Thông tin sản phẩm

#1. Giường Gỗ Mẫu...
GIƯỜNG TRUYỀN THỐNG

Phi vận chuyển:

Thành tiền:

Thông tin đơn hàng

Mã đơn hàng: 1
Loại dịch vụ: Tùy chỉnh
Ngày đặt: 30/04/2025 08:00
Số lượng sản phẩm: 1
Yêu cầu lắp đặt bởi xưởng: Không cần lắp đặt
Ghi chú: Giao hàng nhanh giúp em

Thanh toán

Chỉ định thanh toán

Thanh toán bằng ví
 Thanh toán qua cổng thanh toán

Xác nhận thông tin

Tôi đã kiểm tra thông tin và xác nhận thao tác

Đóng **Thanh toán**

WDCRPB Woodworking platform

Trang chủ Sản phẩm Thiết kế Xưởng mộc Sửa chữa / bảo hành

Đánh giá

Chi tiết đơn #1 Đã hoàn tất

Menu khách hàng

- Trang chủ
- Đơn hàng
- BH & Sửa chữa
- Ví
- Khiếu nại
- Hỗ trợ

Tiến độ đơn hàng

Số	Mô tả	Thời gian
1	Đang chờ xưởng mộc xác nhận đơn hàng	30/04/2025 08:00
2	Đang chờ khách hàng duyệt lịch hẹn	30/04/2025 08:02
3	Đã duyệt lịch hẹn	30/04/2025 08:03
4	Đang chờ khách hàng duyệt hợp đồng	30/04/2025 08:04
5	Đã duyệt hợp đồng	30/04/2025 08:05
6	Đang gia công	30/04/2025 08:08
7	Đang giao hàng lắp đặt	30/04/2025 08:10
8	Đã hoàn tất	30/04/2025 08:13

Thông tin vận chuyển

Giao hàng bởi bên thứ 3 (GHN)

Địa chỉ giao: 123 Nguyễn Văn Linh, Phường Hòa Minh, Quận Liên Chiểu, Đà Nẵng

Địa chỉ lấy hàng: 123 Nguyễn Văn Linh, Phường Tăng Nhơn Phú A, Thành Phố Thủ Đức, Hồ Chí Minh

Đơn vị vận chuyển: Giao hàng nhanh (GHN)

Mã vận đơn: LB9XDX

Tracau

Ngày giao dự kiến: 04/05/2025

Trạng thái vận chuyển: Đơn hàng vừa được tạo

- Click “Thanh toán và xác nhận đơn hàng”
- Choose payment method and process payment

3.2.3 Guarantee Service Booking Feature

3.2.3.1 Customer request a guarantee order

Yêu cầu sửa chữa / bảo hành sản phẩm

Chọn đơn hàng đã hoàn thành

Địa chỉ giao hàng

Gửi yêu cầu sửa chữa

Địa chỉ

Nhận thông tin mới từ chúng tôi

Theo dõi chúng tôi

Dai hoc FPT
www@com.vn

Đăng ký bằng cách gửi email để nhận được những thông tin mới nhất từ chúng tôi

Email*

Facebook
Instagram

WDCRPB

wdcrbp.vercel.app/guarantee

Trang chủ Sản phẩm Thiết kế Xưởng mộc Sửa chữa / bảo hành

Yêu cầu sửa chữa / bảo hành sản phẩm

Chọn đơn hàng đã hoàn thành

Đơn #1 - 30/04/2025 - 5.605.000 ₫

Địa chỉ giao hàng

Quản lý địa chỉ

123 Nguyễn Văn Linh, Phường Hòa Minh, Quận Liên Chiểu, Đà Nẵng

Sản phẩm cần sửa chữa / bảo hành

Gửi yêu cầu sửa chữa

Chọn sản phẩm cần sửa chữa / bảo hành

Mộc Chạm

Địa chỉ xưởng: 123 Nguyễn Văn Linh, Phường Tăng Nhơn Phú A, Thành Phố Thủ Đức, Hồ Chí Minh

Loại hình kinh doanh: Hộ kinh doanh

Loại xưởng: Xưởng vàng

Xem xưởng

Địa chỉ

Đại học FPT
ww@com.vn

Nhận thông tin mới từ chúng tôi

Đăng ký bằng cách gửi email để nhận được những thông tin mới nhất từ chúng tôi

Email*

Theo dõi chúng tôi

Facebook
Instagram

WDCRPB

wdcrbp.vercel.app/guarantee

Trang chủ Sản phẩm Thiết kế Xưởng mộc Sửa chữa / bảo hành

Yêu cầu sửa chữa / bảo hành sản phẩm

Chọn đơn hàng đã hoàn thành

Đơn #1 - 30/04/2025 - 5.605.000 ₫

Chọn sản phẩm cần sửa chữa / bảo hành

#1. Giường Gỗ Mẫu Hoàng Gia Họa Lá Tây

CÔNG HÀN

Thời hạn bảo hành: 12 tháng

Hết hạn ngày: 30/04/2026

Đóng

Sản phẩm cần sửa chữa / bảo hành

Gửi yêu cầu sửa chữa

Chọn sản phẩm cần sửa chữa / bảo hành

Mộc Chạm

Địa chỉ xưởng: 123 Nguyễn Văn Linh, Phường Tăng Nhơn Phú A, Thành Phố Thủ Đức, Hồ Chí Minh

Loại hình kinh doanh: Hộ kinh doanh

Loại xưởng: Xưởng vàng

Xem xưởng

Địa chỉ

Đại học FPT
ww@com.vn

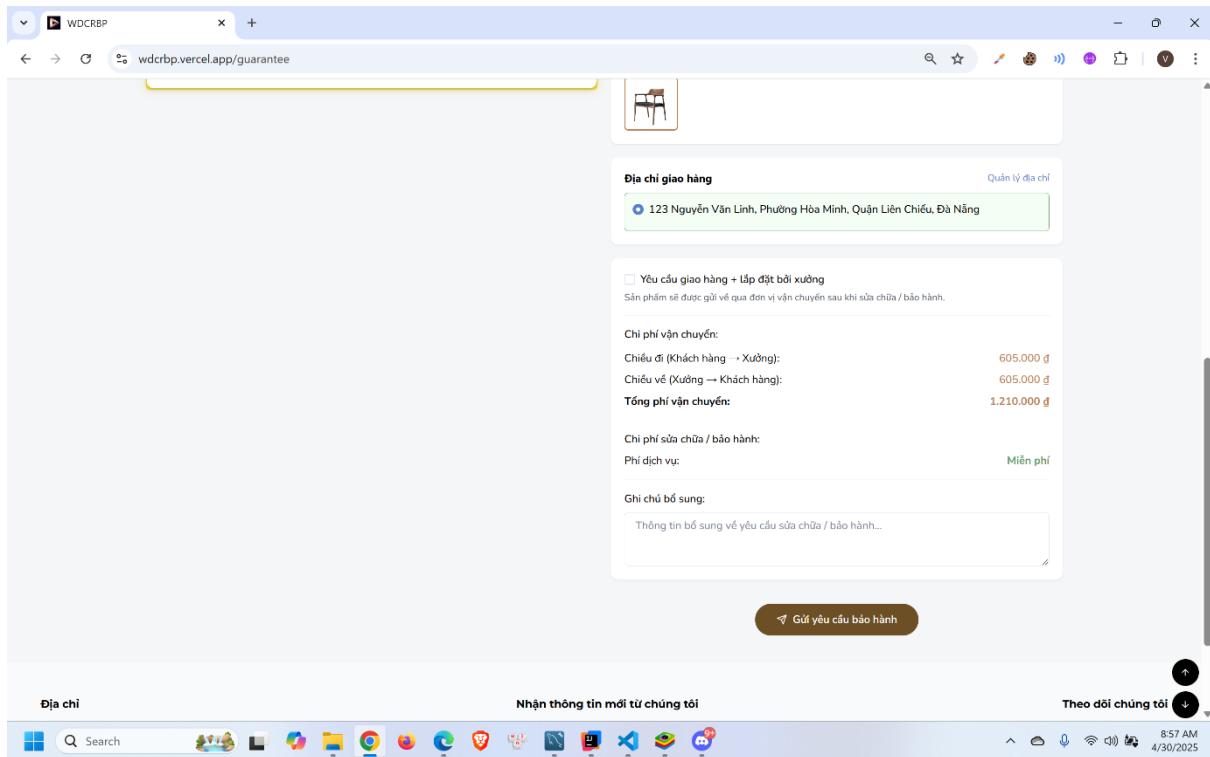
Nhận thông tin mới từ chúng tôi

Đăng ký bằng cách gửi email để nhận được những thông tin mới nhất từ chúng tôi

Email*

Theo dõi chúng tôi

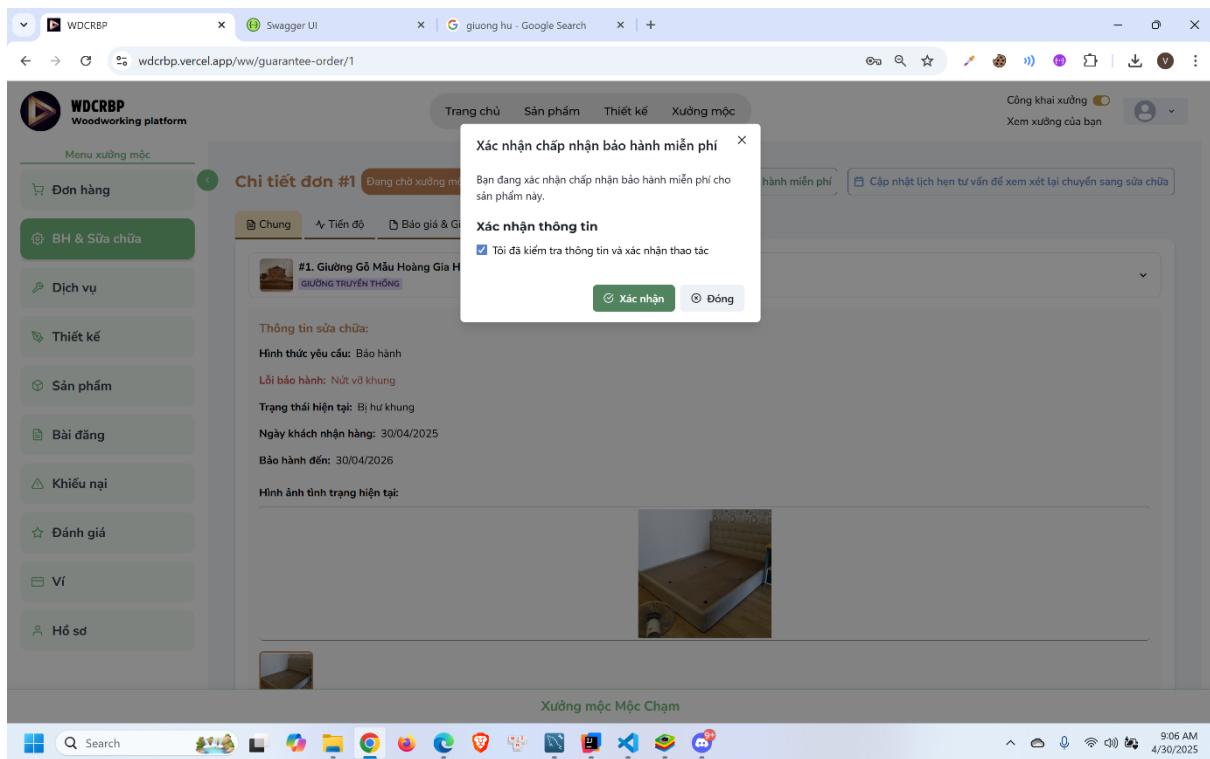
Facebook
Instagram



1. Click “Sửa chữa / bảo hành” in the menu
2. Select order to request
3. Select product to request
4. Fill in the form and click “Gửi yêu cầu bảo hành”

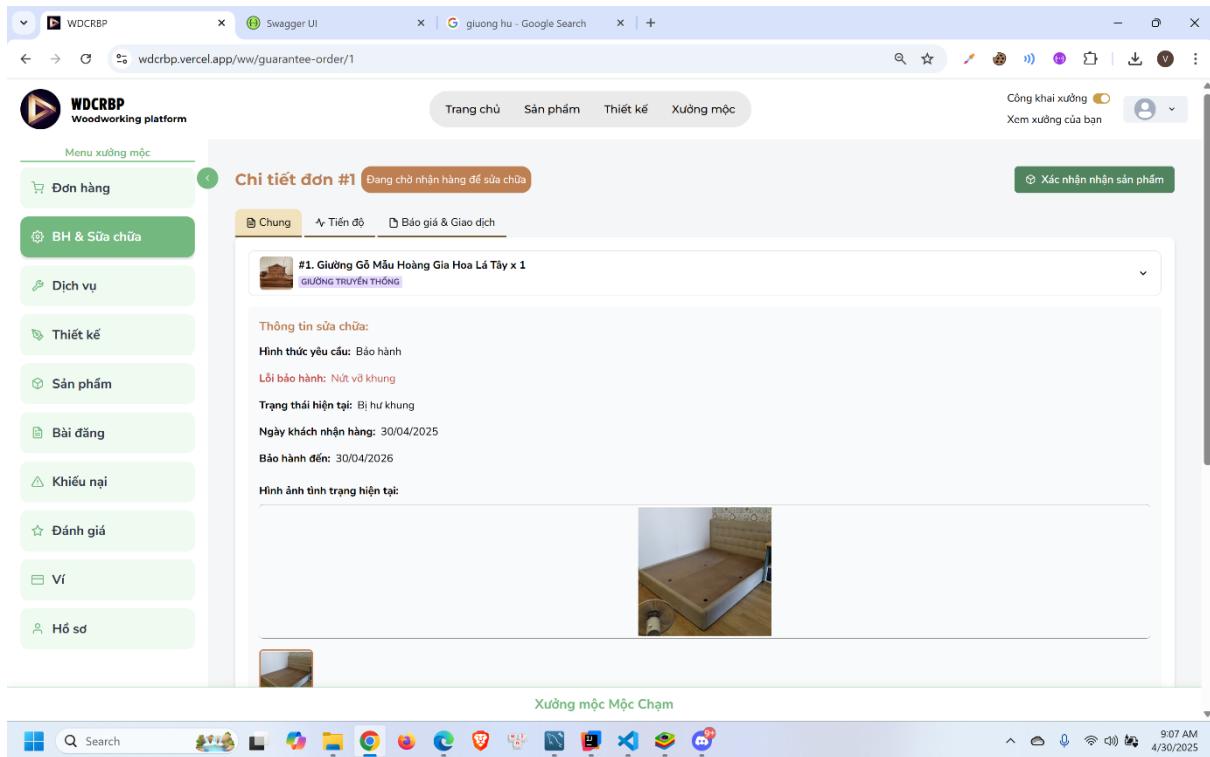
3.2.3.2 Woodworker accept the guarantee order

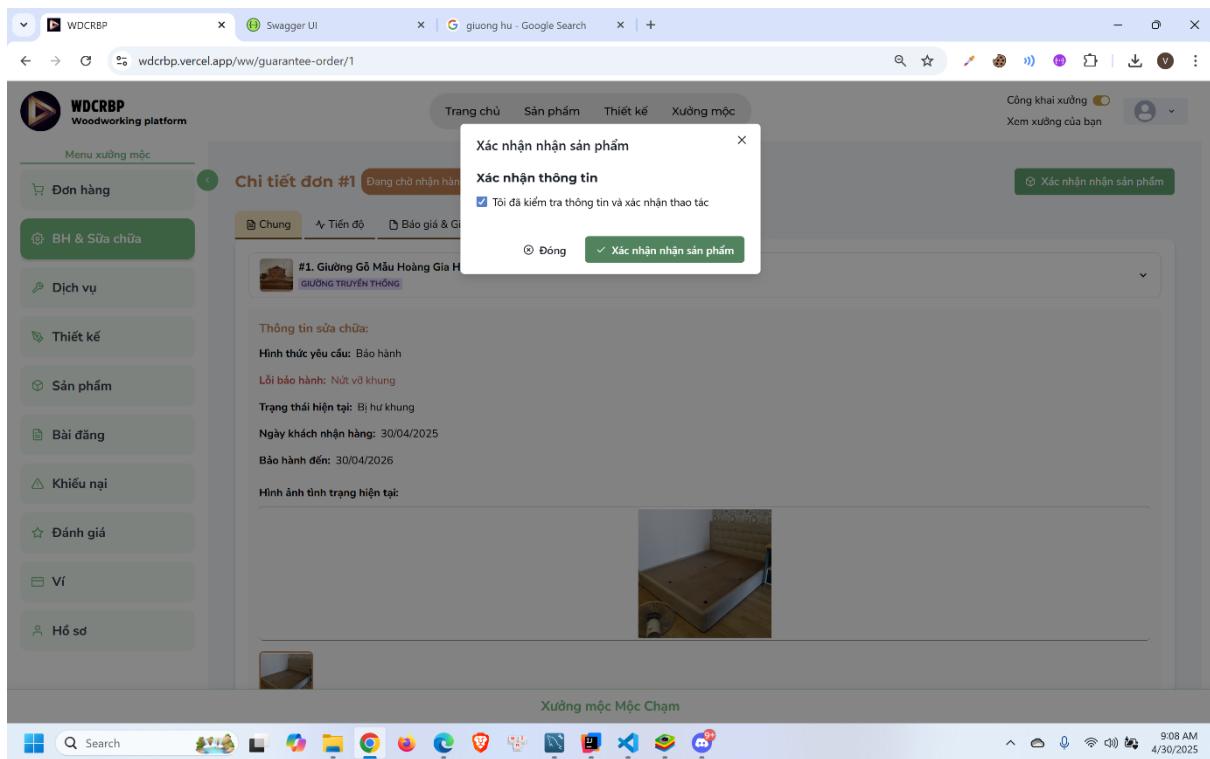
The screenshot shows a detailed view of a repair order. The top navigation bar includes links for 'Trang chủ', 'Sản phẩm', 'Thiết kế', and 'Xưởng mộc'. On the left, a sidebar lists categories: 'Đơn hàng', 'BH & Sửa chữa' (highlighted in green), 'Dịch vụ', 'Thiết kế', 'Sản phẩm', 'Bài đăng', 'Khiếu nại', 'Đánh giá', 'Ví', and 'Hỗ trợ'. The main content area shows a repair order for 'Giường Gỗ Mẫu Hoàng Gia Họa Lá Tây x 1'. It includes sections for 'Thông tin sửa chữa', 'Lỗi bảo hành', 'Trạng thái hiện tại', and 'Ngày khách nhận hàng'. A photo of the bed frame is shown. At the bottom right, a green button reads 'Xưởng mộc Mộc Chạm'.



1. Click “Chấp nhận bảo hành miễn phí”
2. Click “Xác nhận”

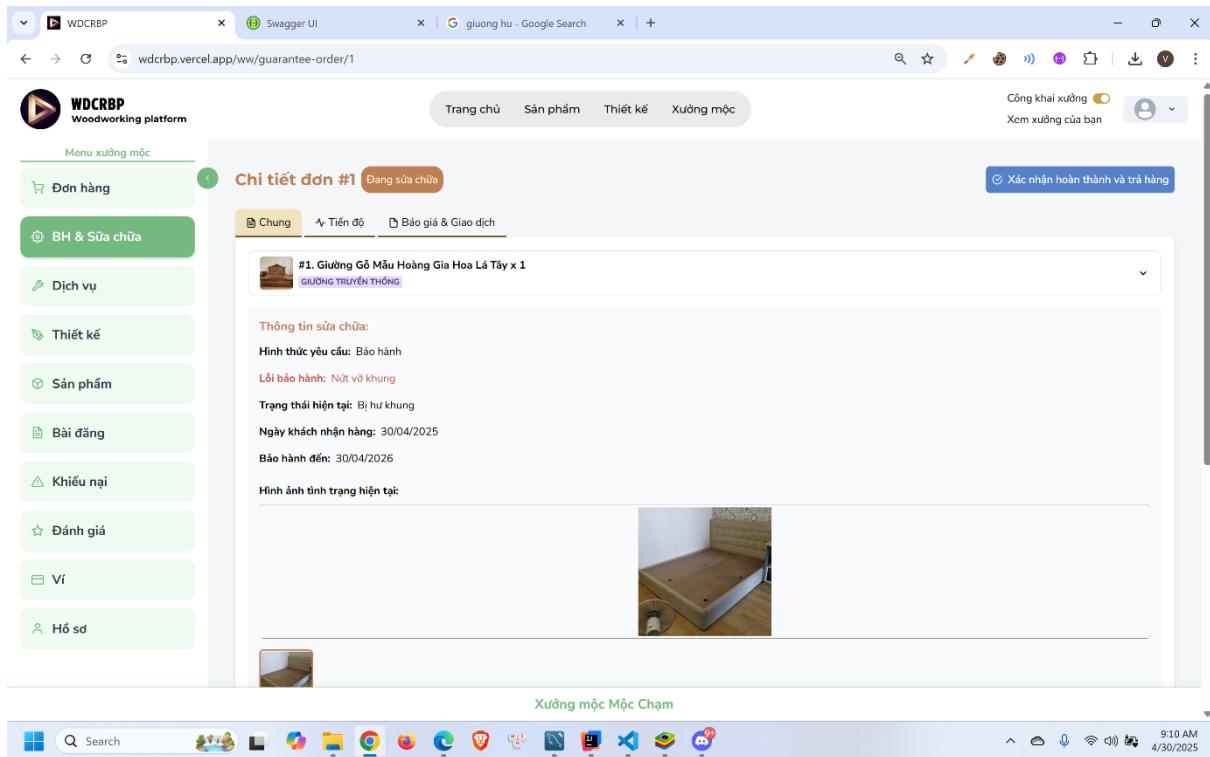
3.2.3.3 Woodworker confirm receive product

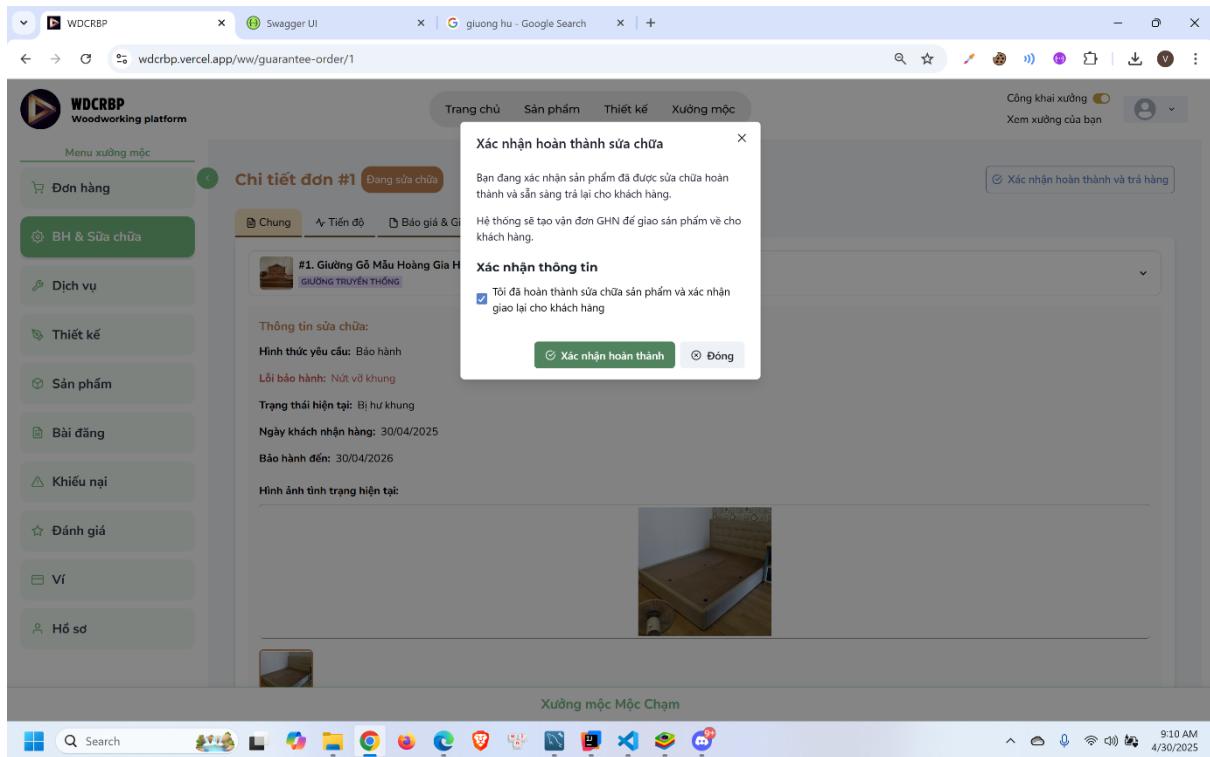




1. Click “Xác nhận nhận sản phẩm”
2. Click “Xác nhận”

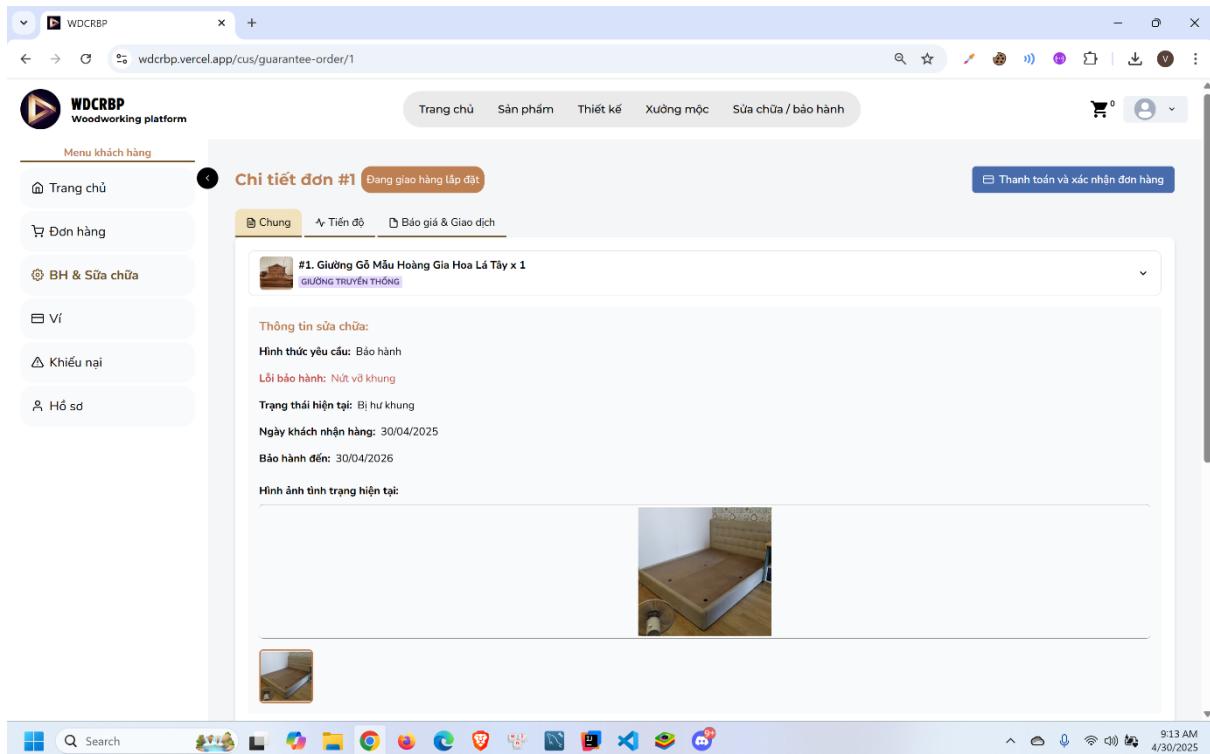
3.2.3.4 Woodworker confirm repair finish

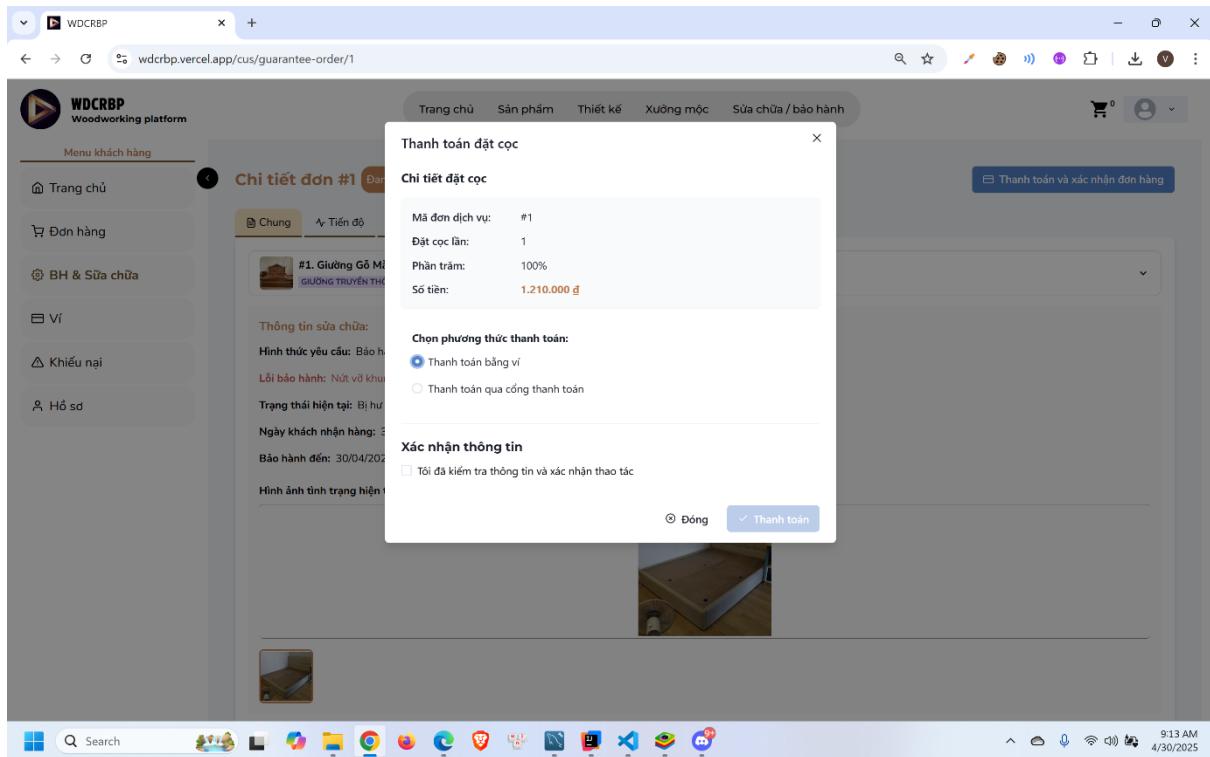




1. Click “Xác nhận hoàn thành và trả hàng”
2. Click “Xác nhận hoàn thành”

3.2.3.5 Customer confirm receive product and process payment



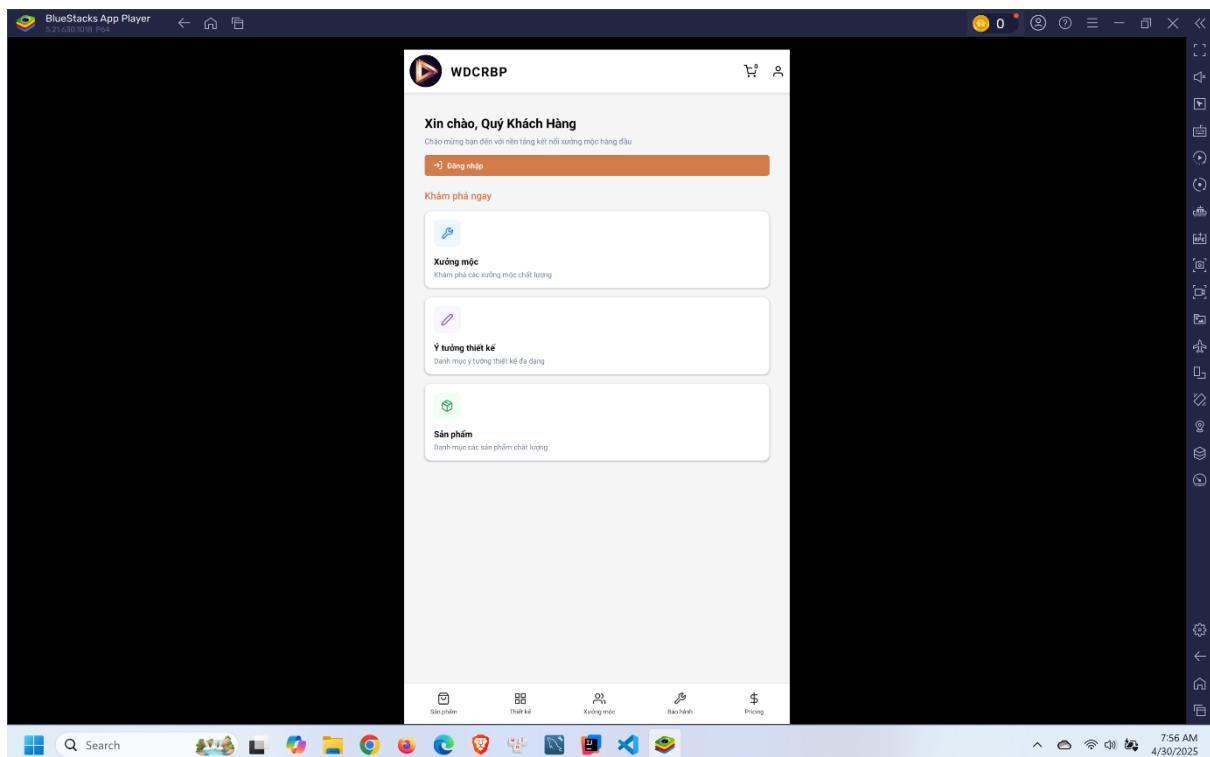


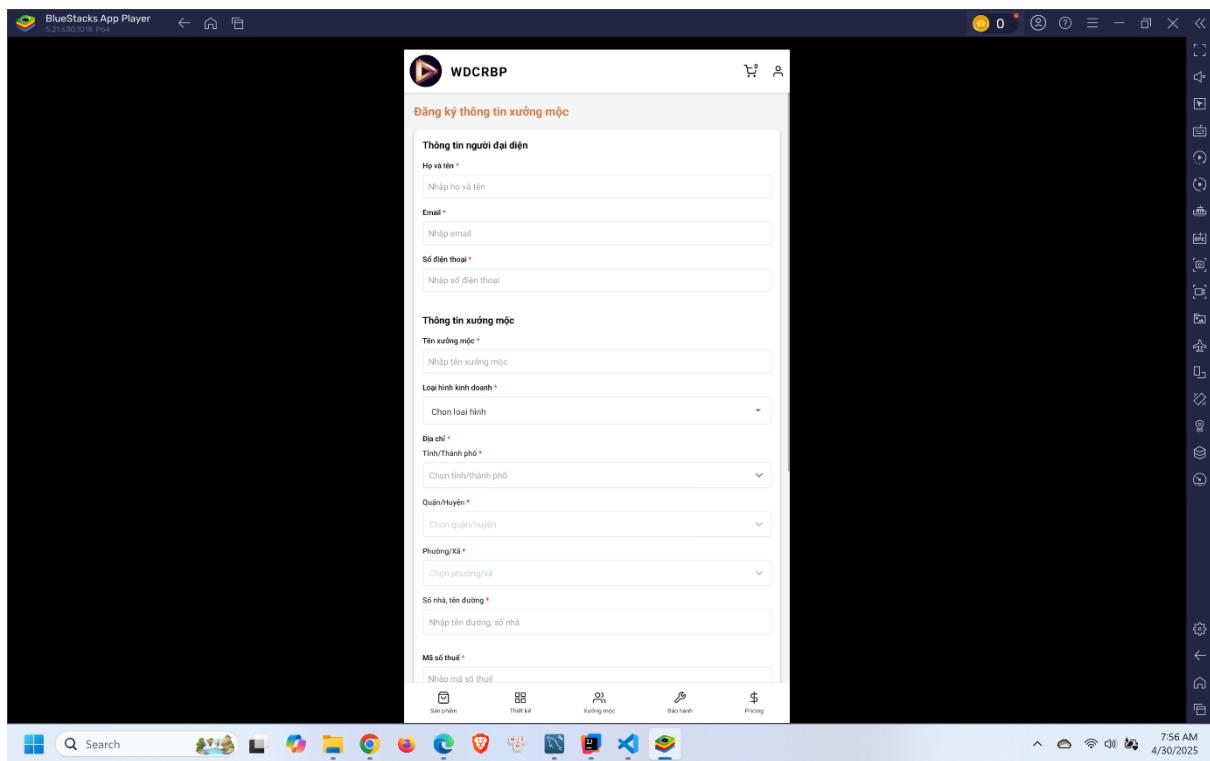
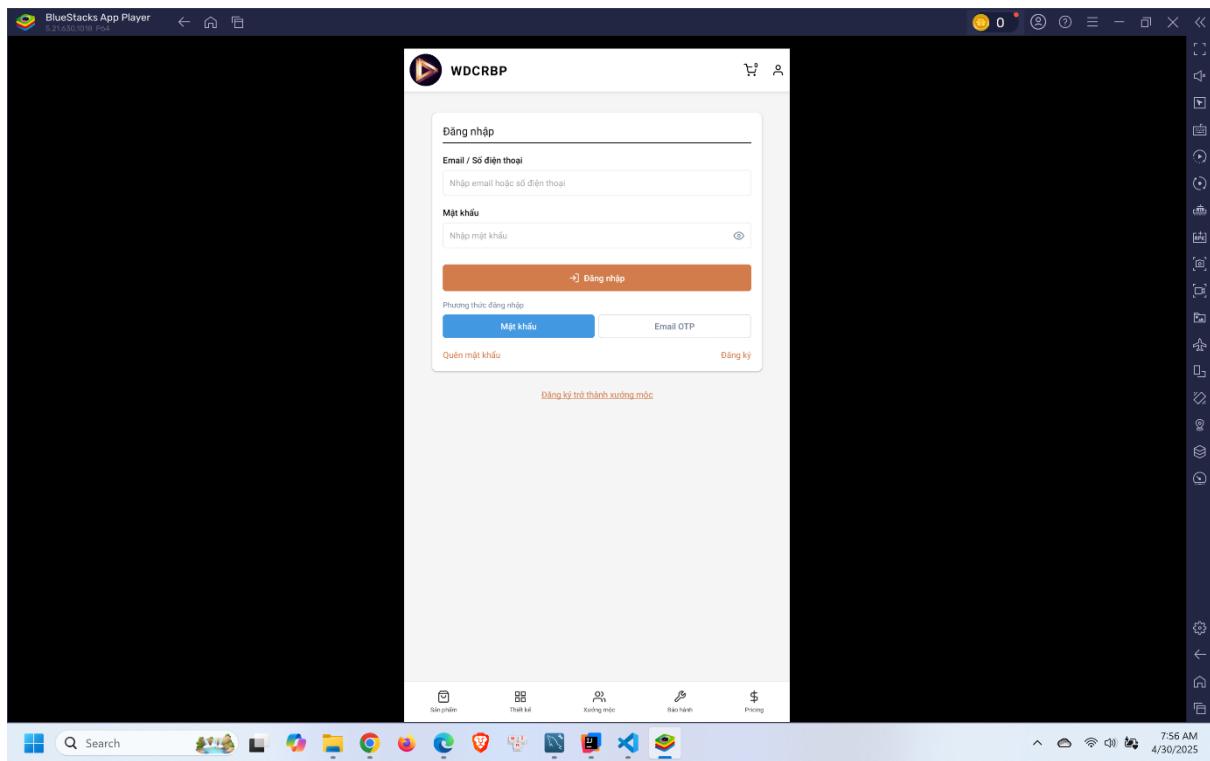
1. Click “Thanh toán và xác nhận đơn hàng”
2. Choose payment method and process payment

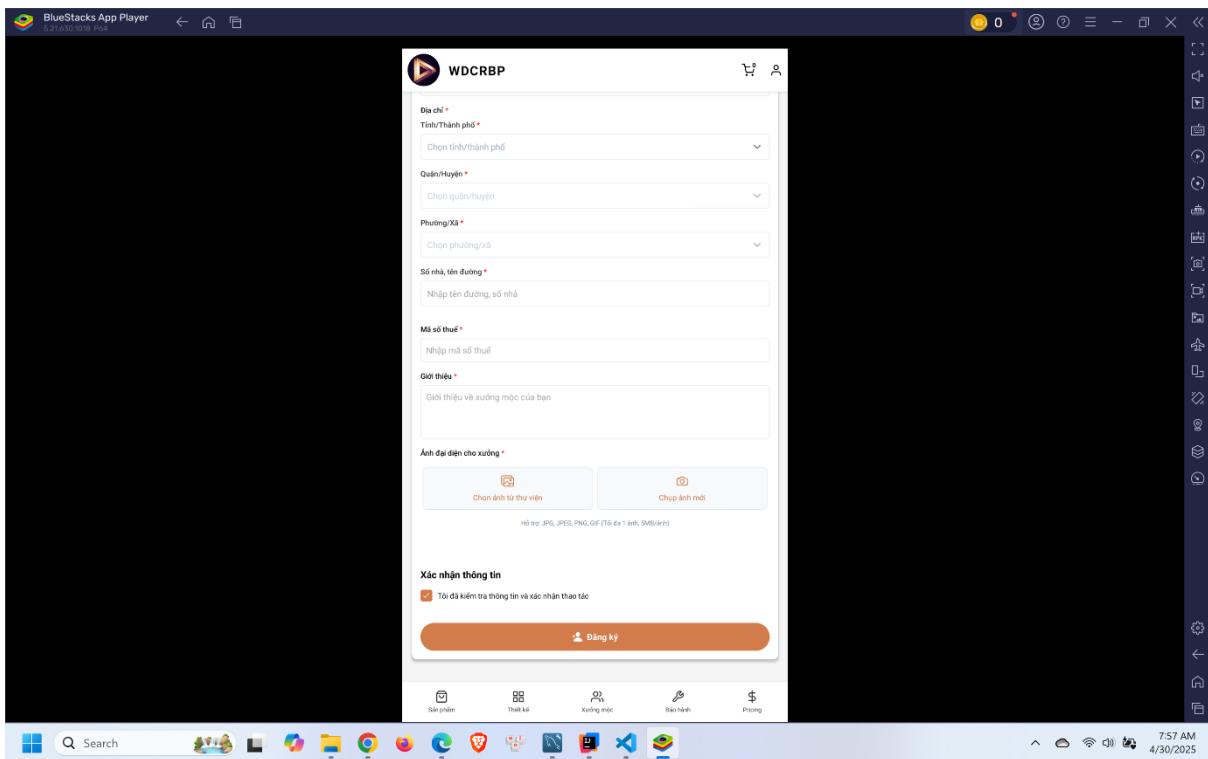
3.3 Mobile application

3.3.1 Woodworker Registration Feature

3.3.1.1 Guest register as a woodworker

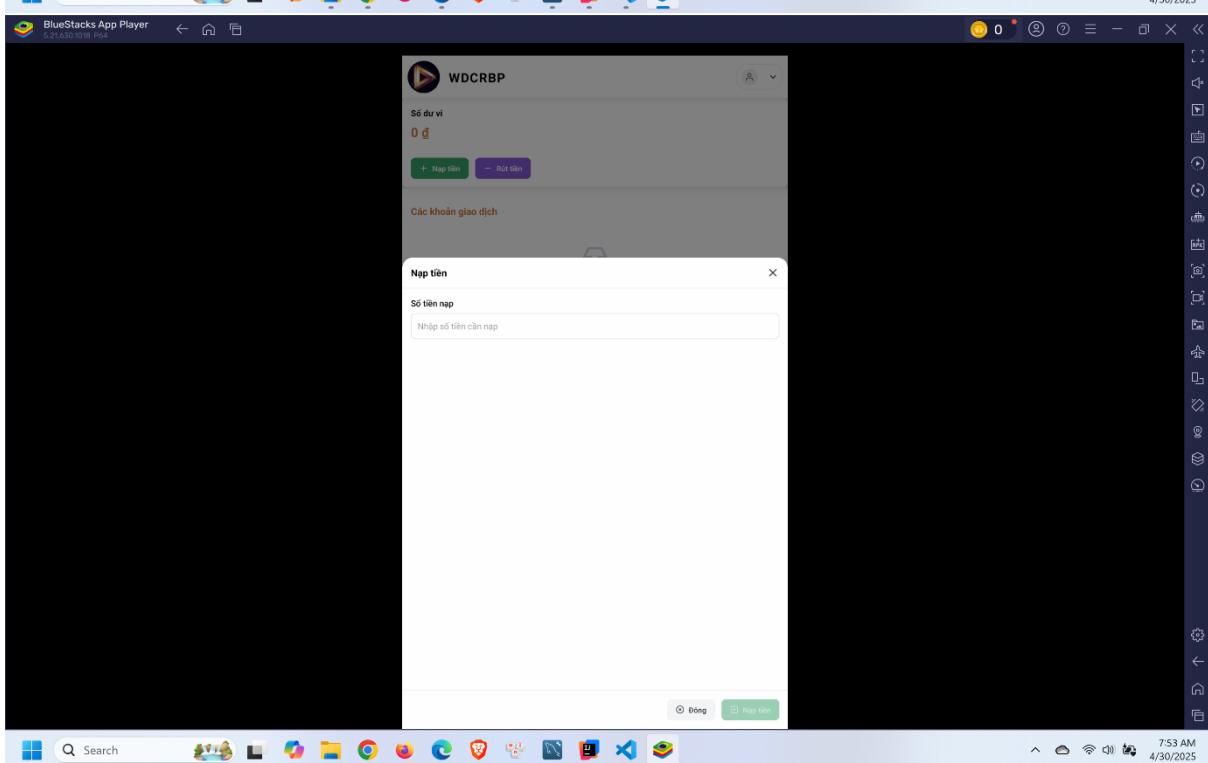
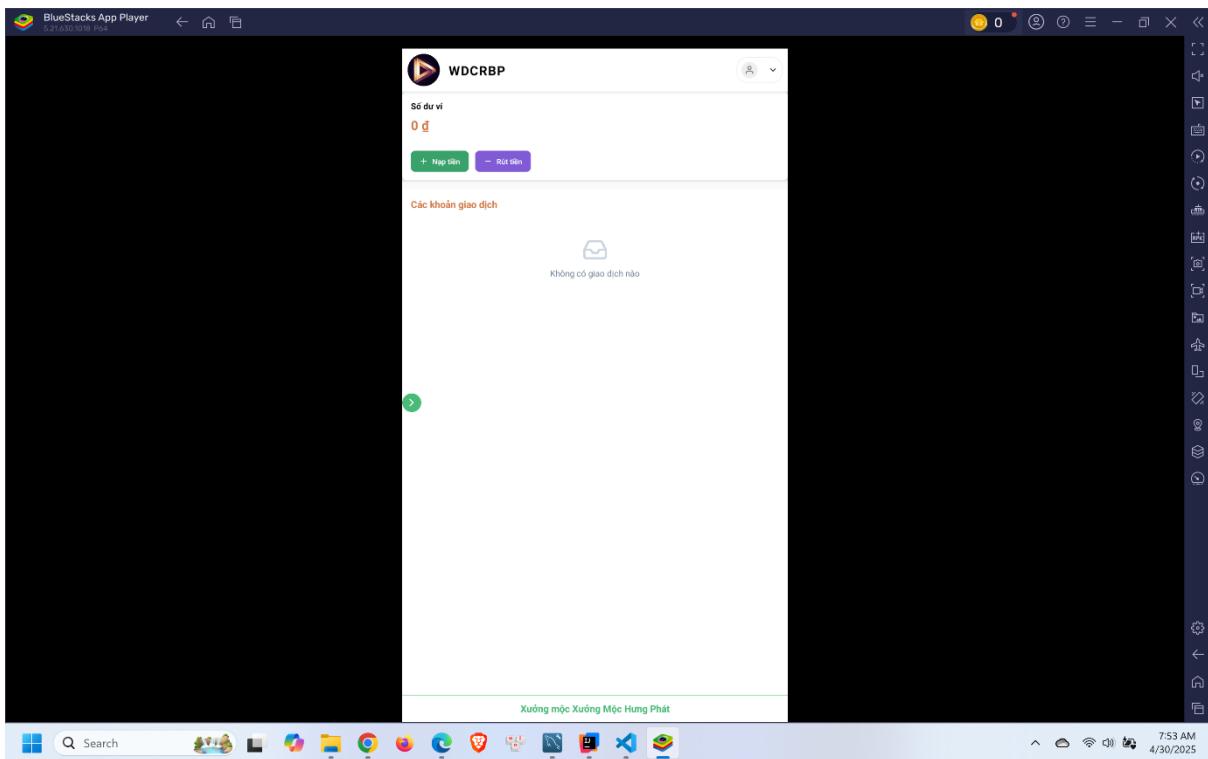


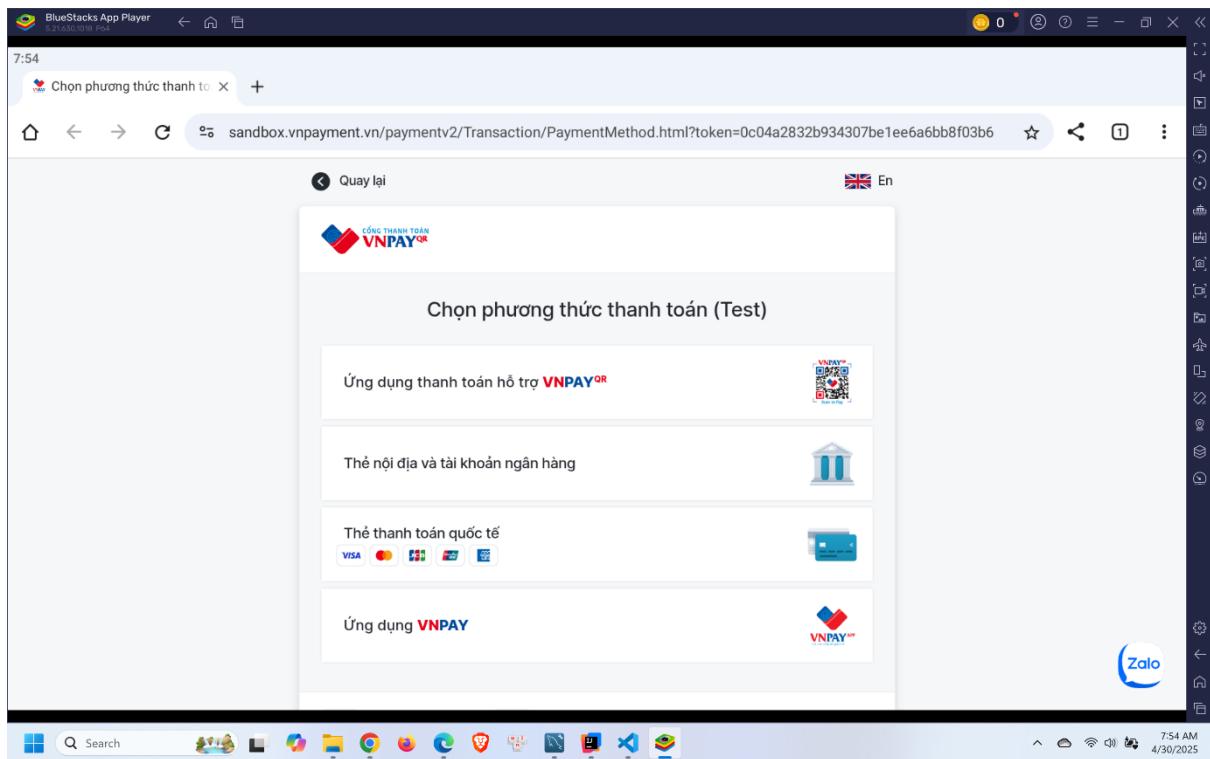




1. Click account icon in the header
2. Click “Đăng ký trở thành xưởng mộc”
3. Fill in the registration form
4. Click “Đăng ký”

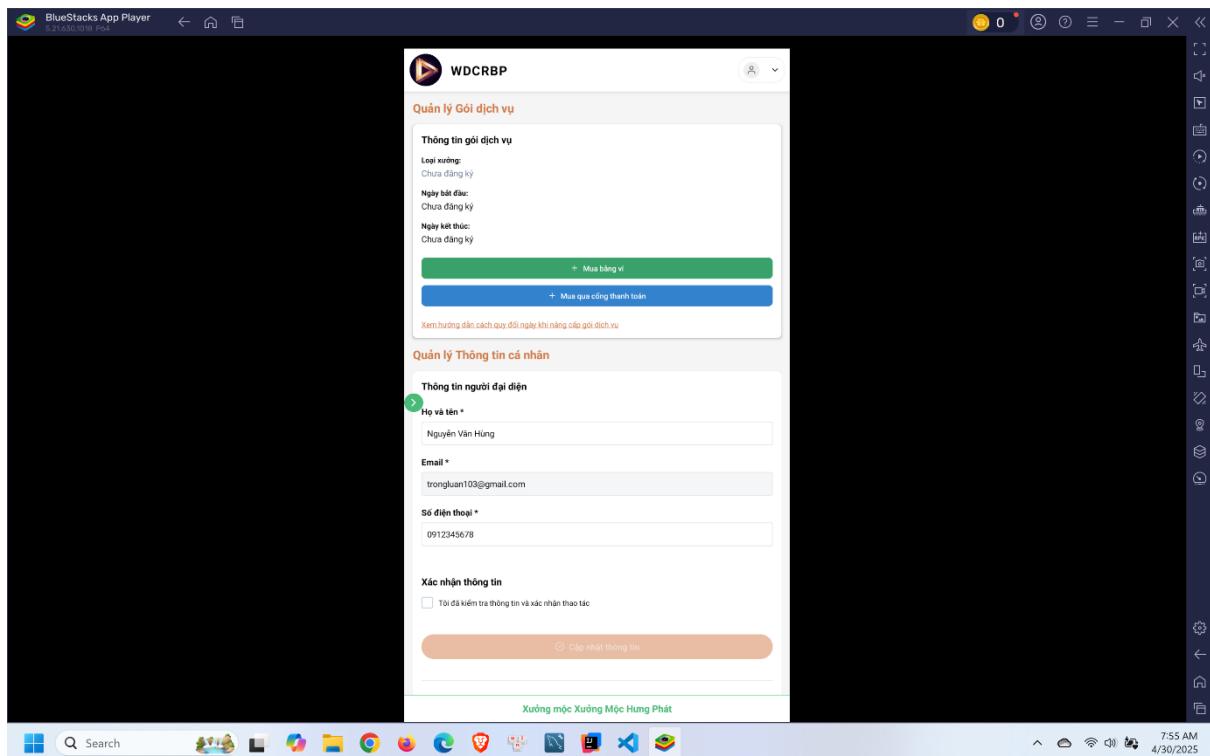
3.3.1.2 Woodworker deposit money to wallet

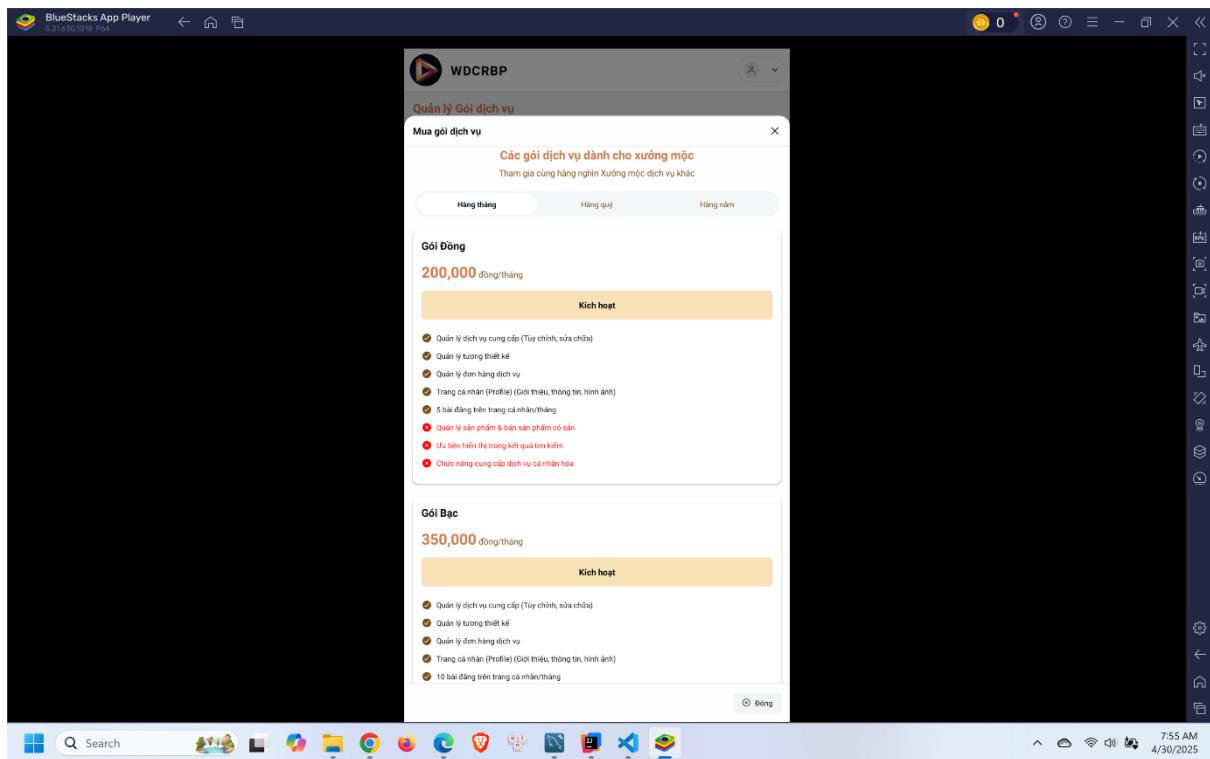




1. Click “Nạp tiền”
2. Enter amount and click “Nạp tiền”

3.3.1.3 Woodworker upgrade service pack

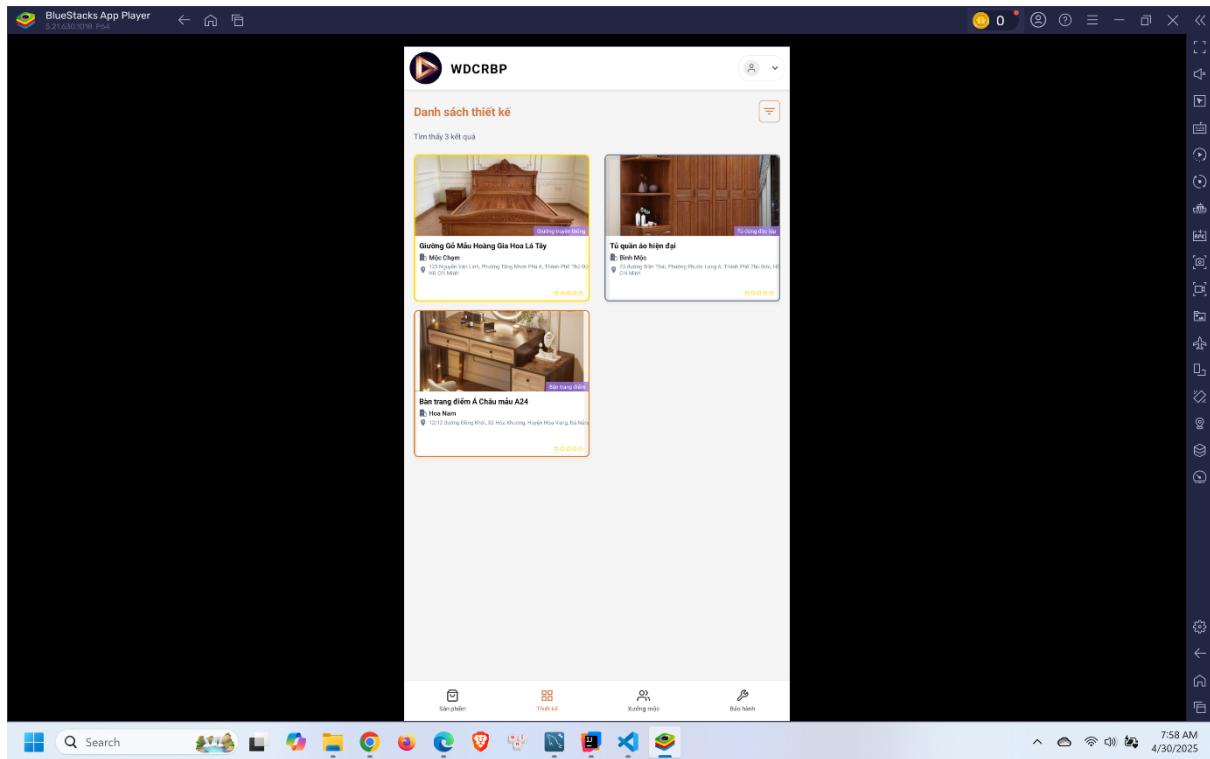


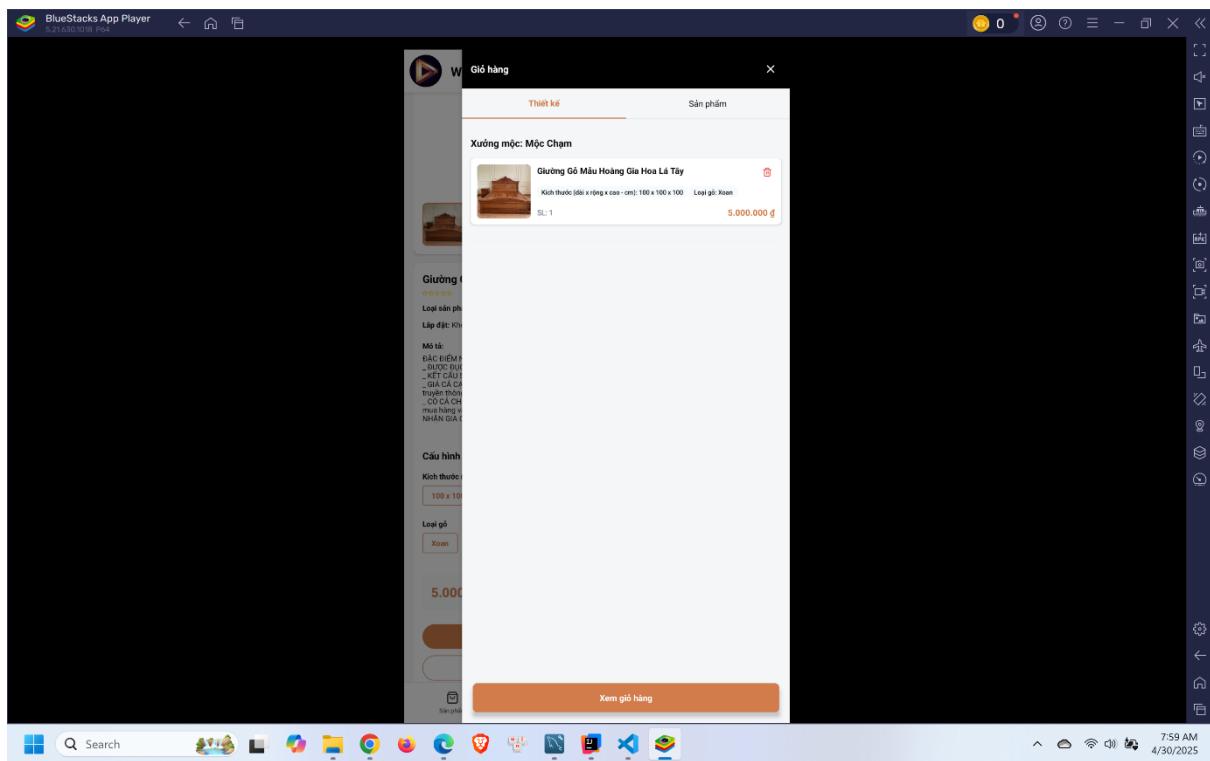
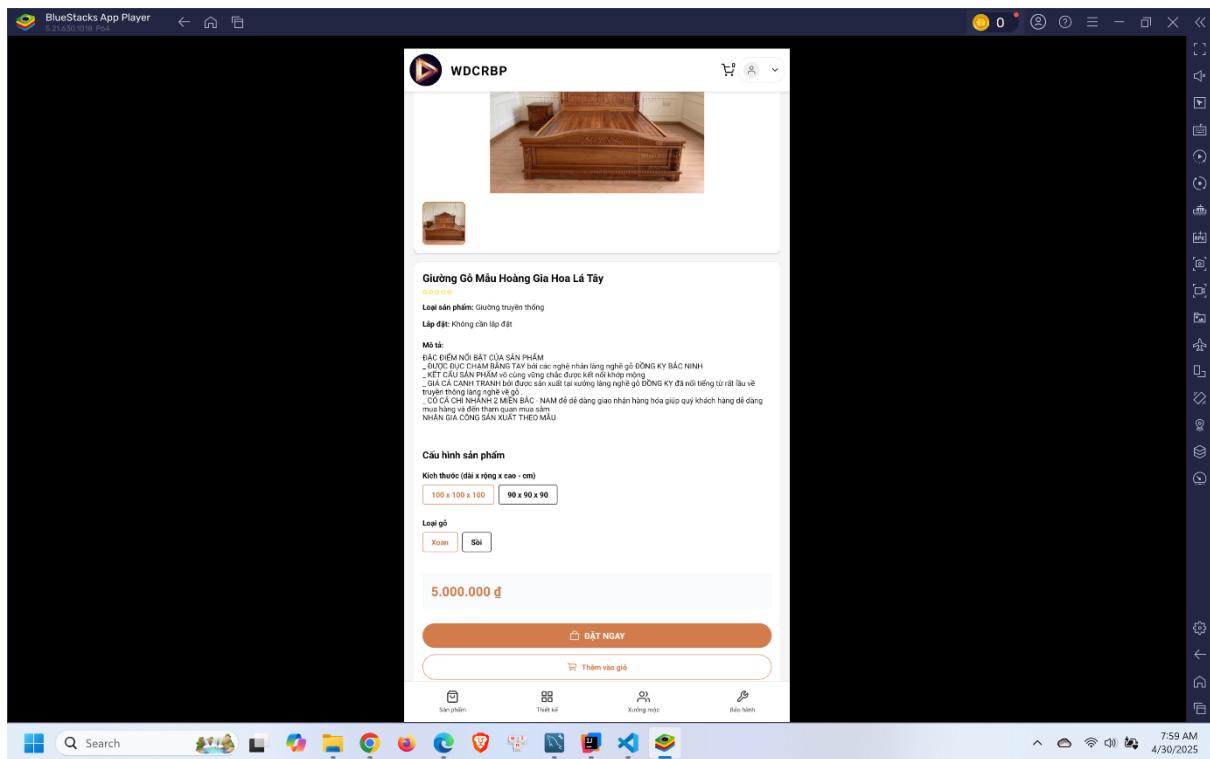


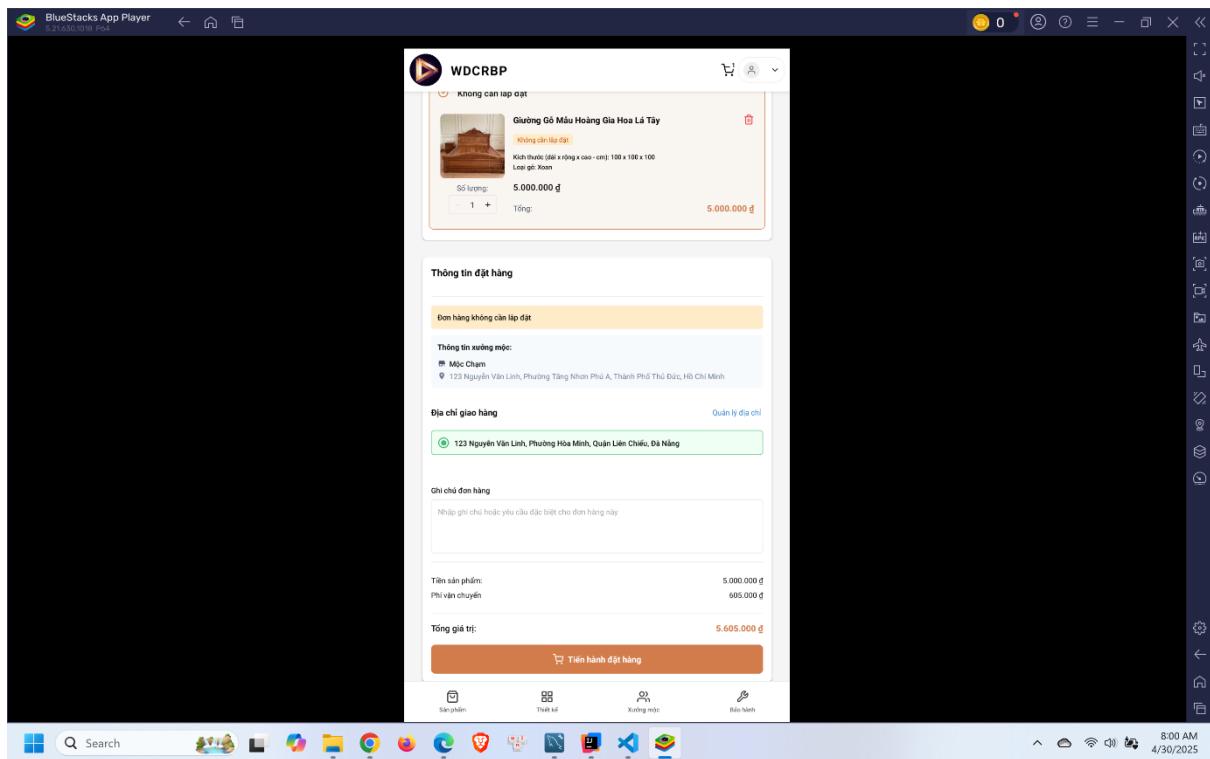
1. Click “Mua bằng ví” or “Mua qua công thanh toán”
2. Choose pack to use and click “Kích hoạt”

3.3.2 Customization Service Booking Feature

3.3.2.1 Customer request a customization order

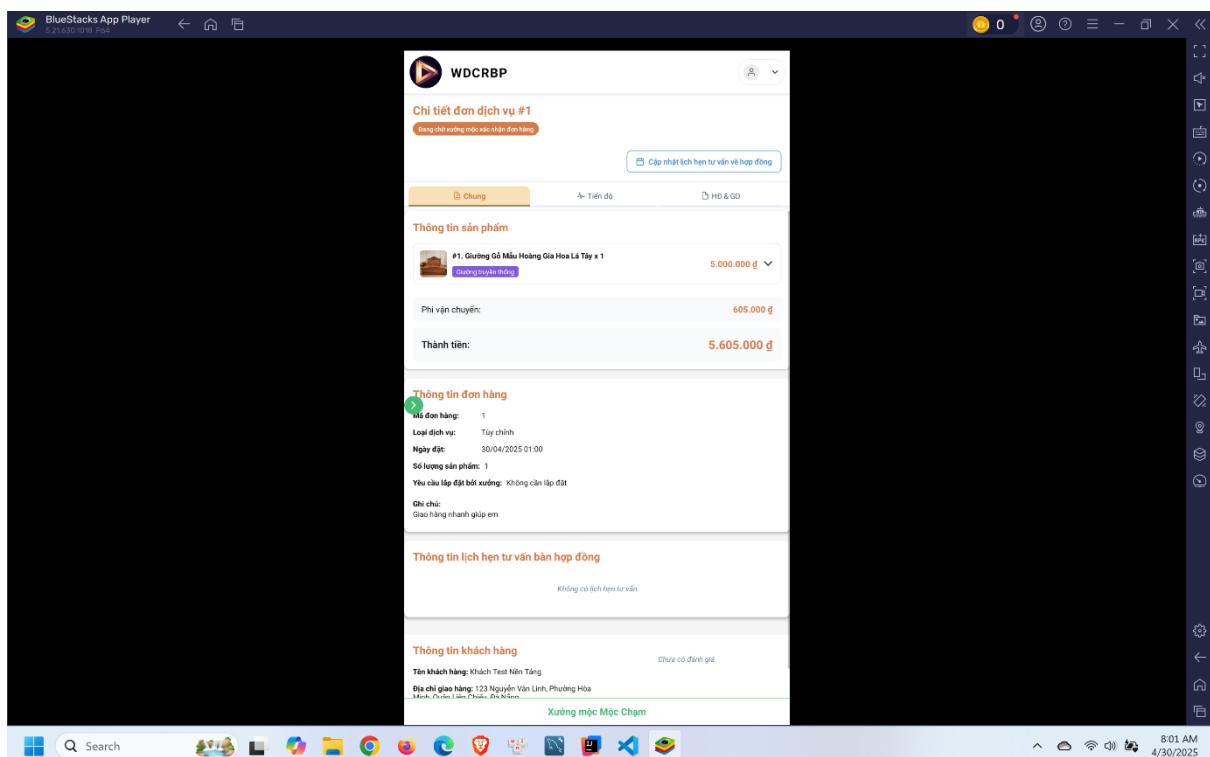


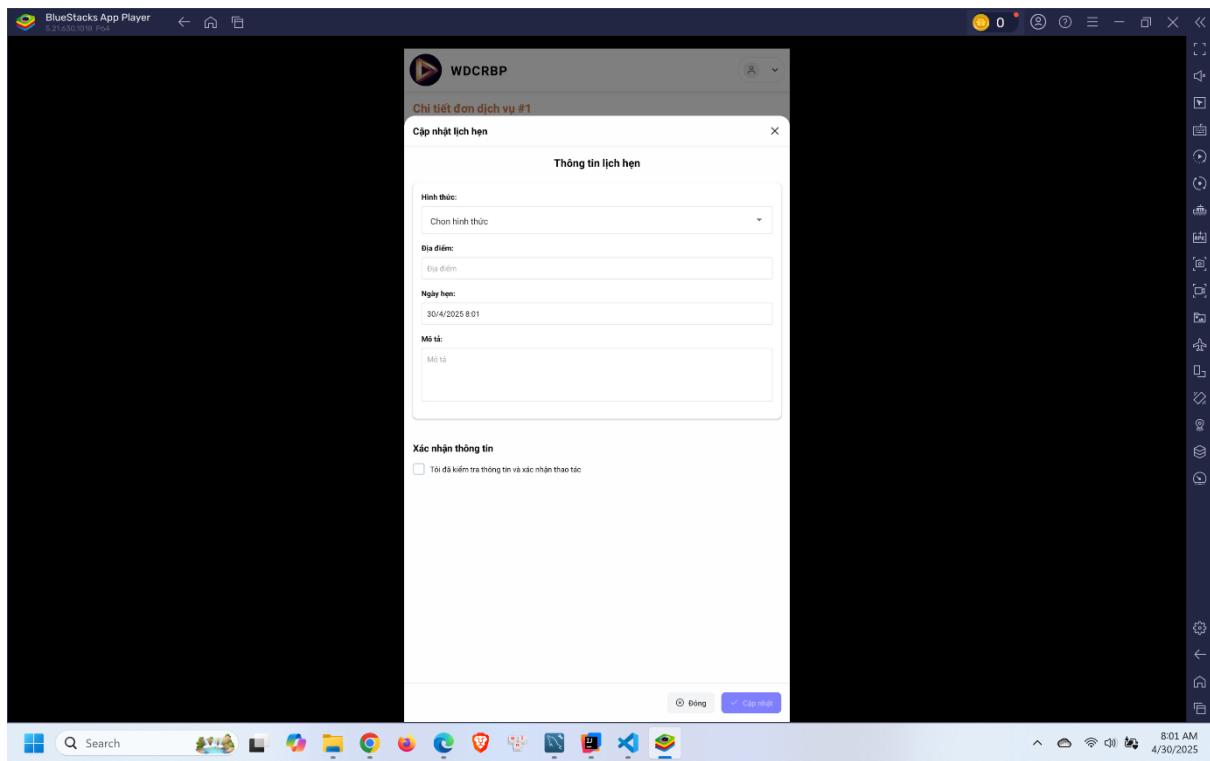




1. Click “Thiết kế” in the menu
2. Select design to view detail
3. Click “Thêm vào giỏ”
4. Click cart icon in the header
5. Click “Xem giỏ hàng”
6. Fill in the form and click “Tiến hành đặt hàng”

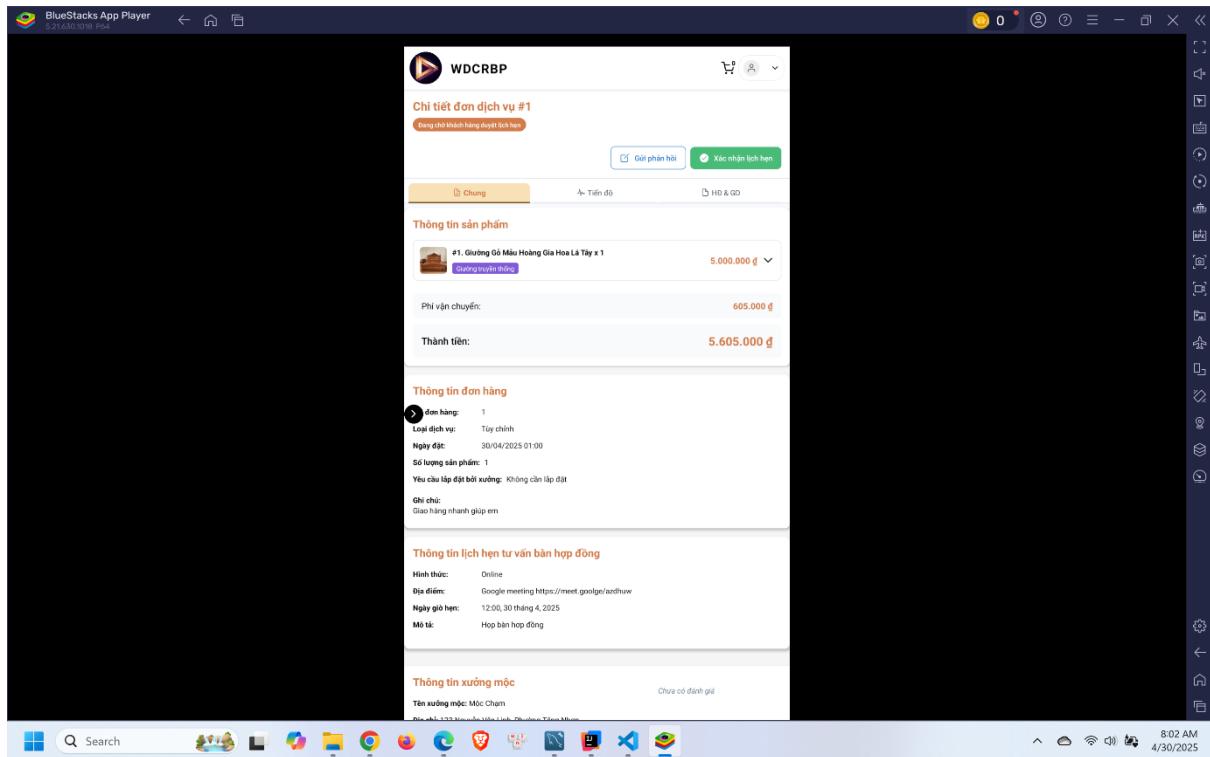
3.3.2.2 Woodworker book a consultant appointment

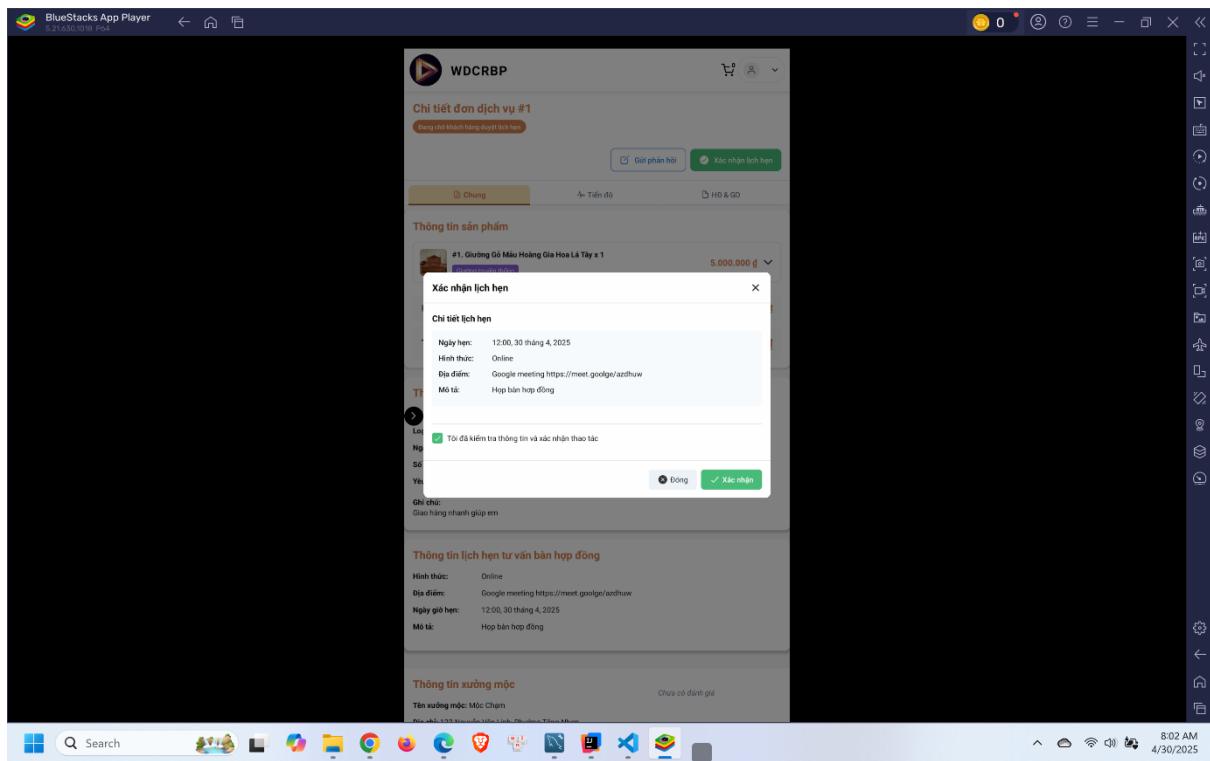




1. Click “Cập nhật lịch hẹn tư vấn về hợp đồng”
2. Enter the form and click “Cập nhật”

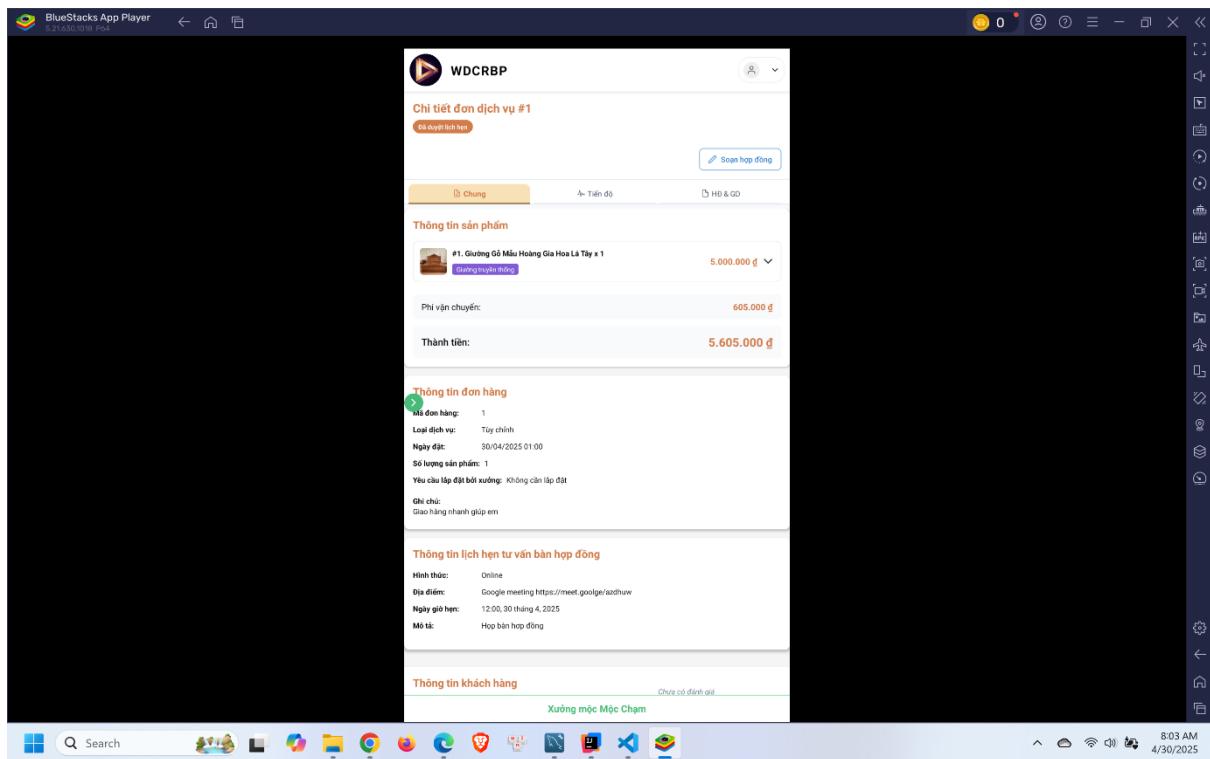
3.3.2.3 Customer confirm consultant appointment

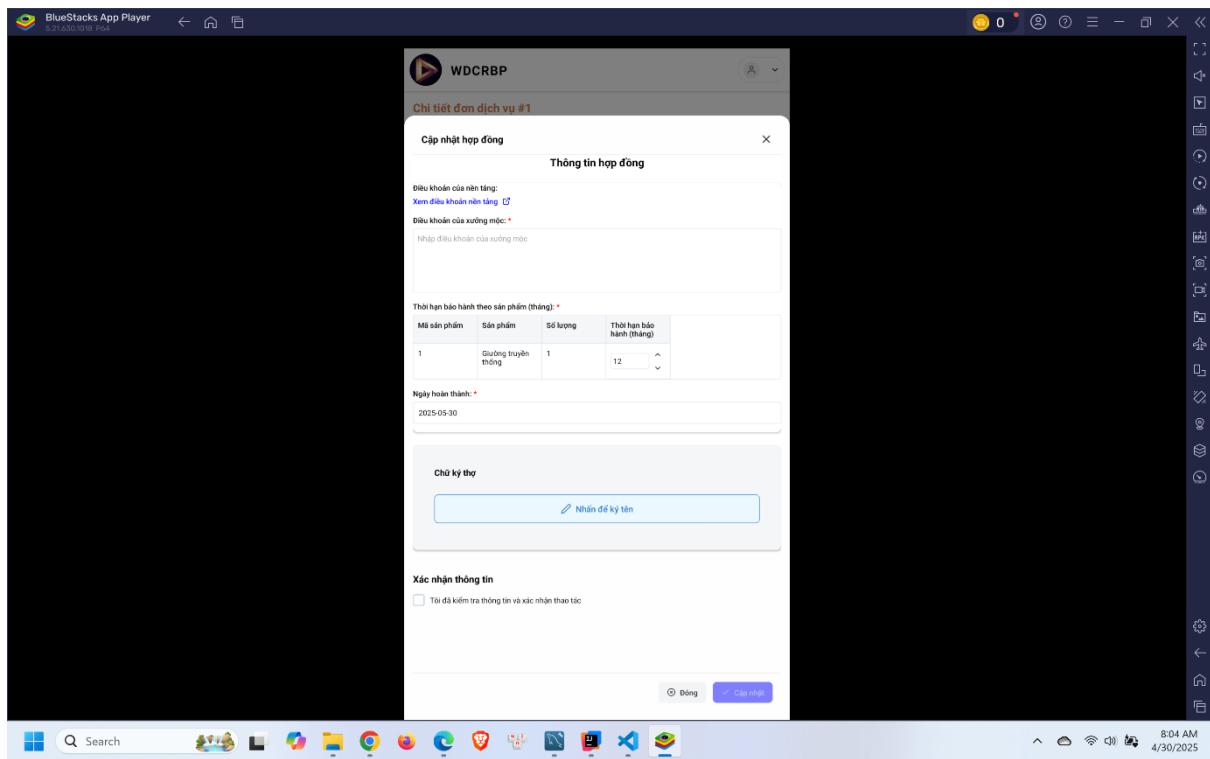




1. Click “Xác nhận lịch hẹn”
2. Click “Xác nhận”

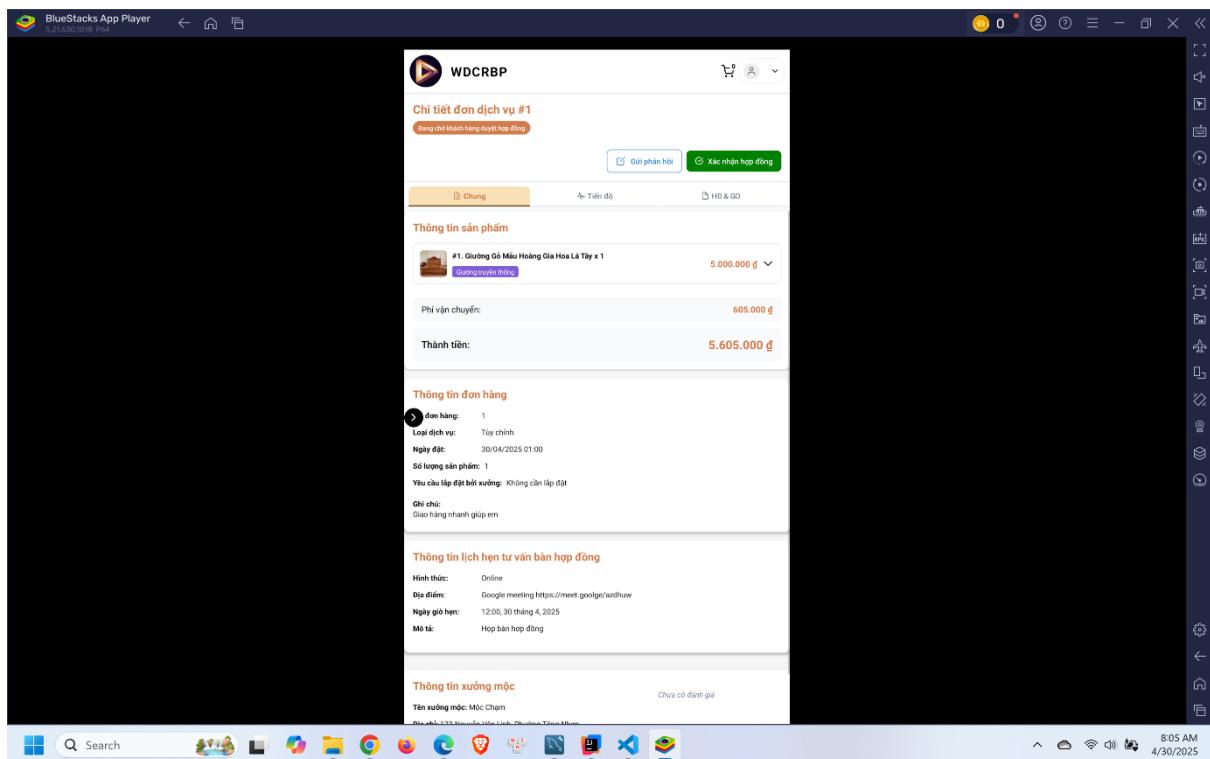
3.3.2.4 Woodworker update the contract

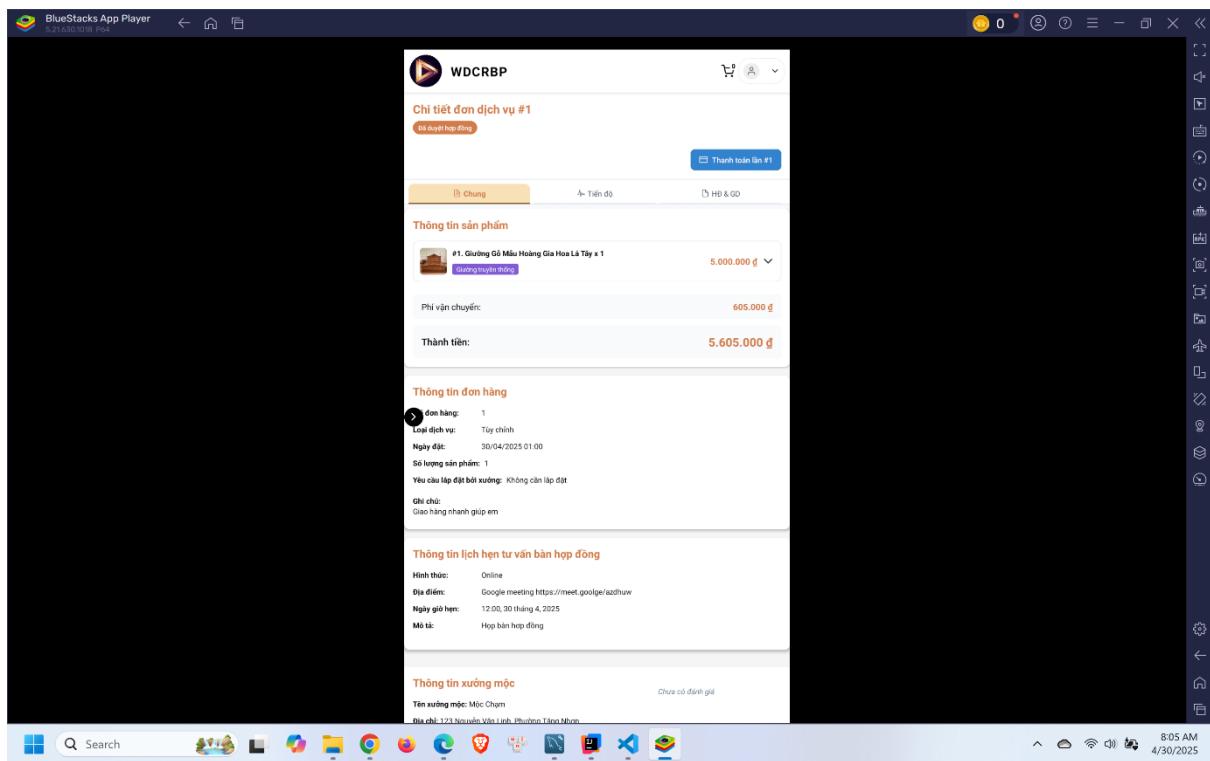
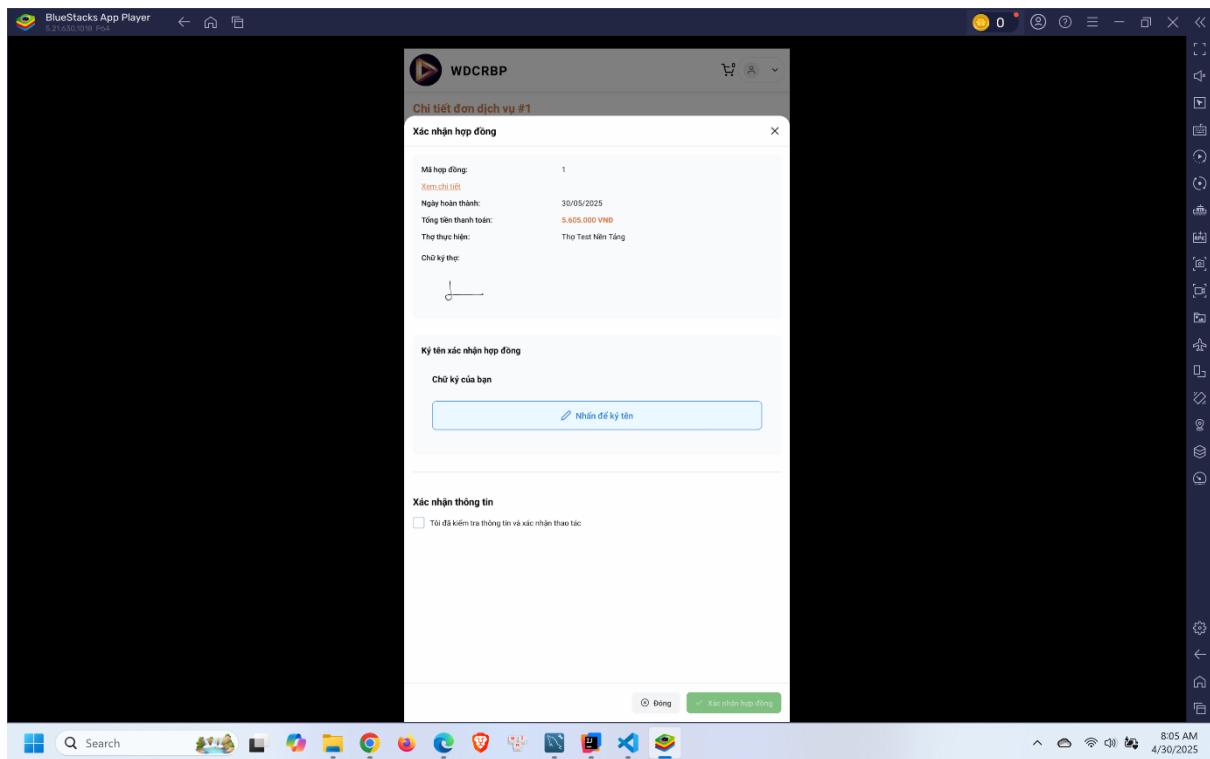


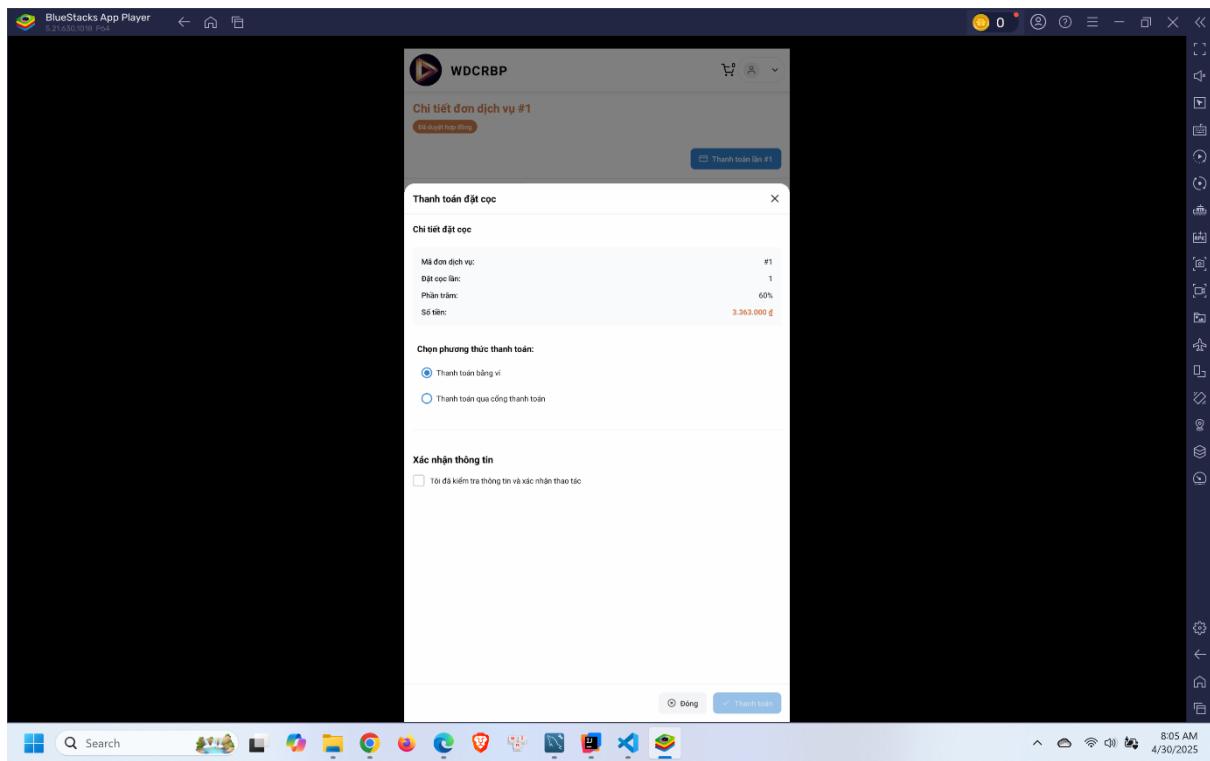


1. Click “Soạn hợp đồng”
2. Enter the form and sign in the box
3. Click “Cập nhật”

3.3.2.5 Customer confirm contract and process payment

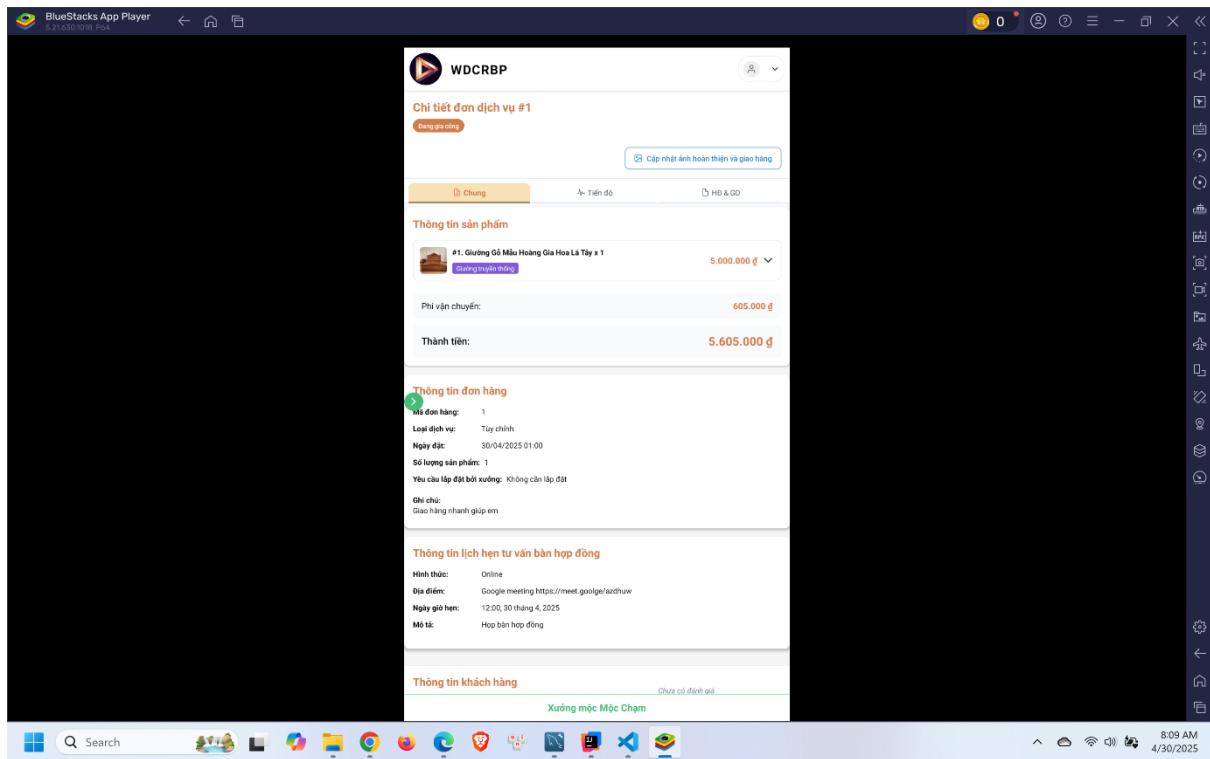


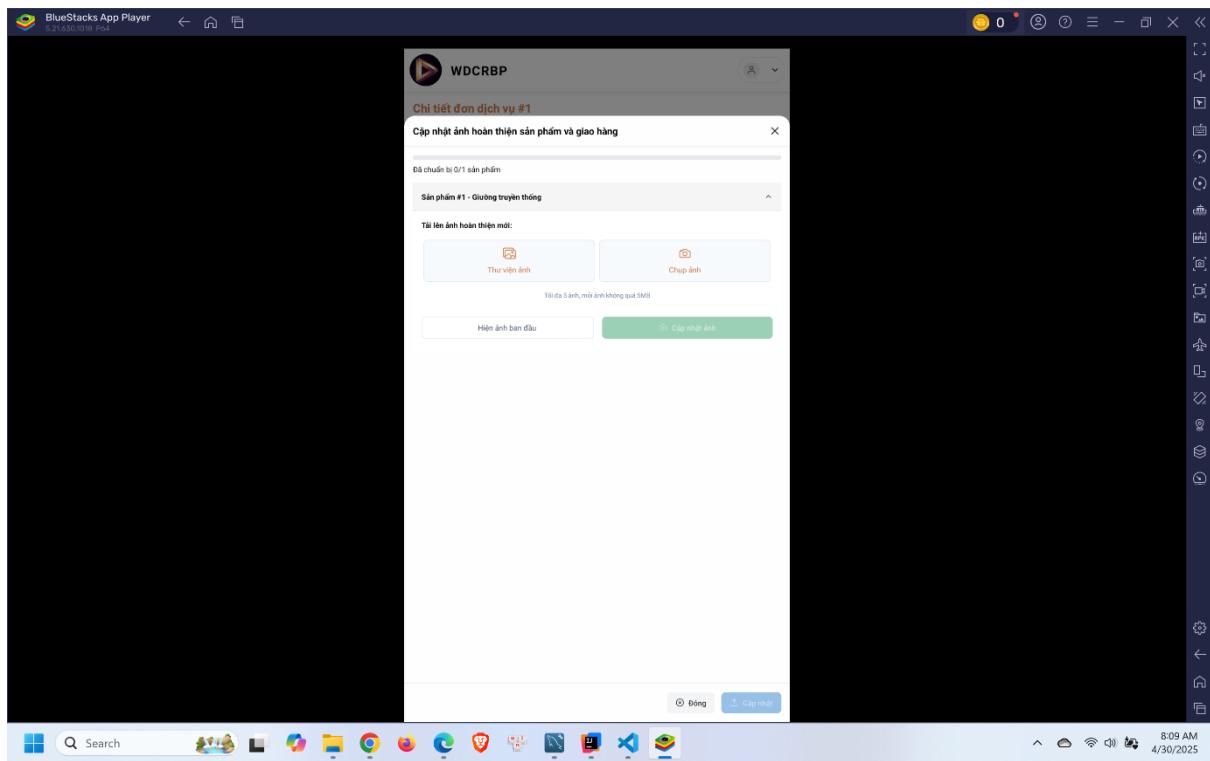




1. Click “Xác nhận hợp đồng”
2. Check information and sign in the box
3. Click “Xác nhận hợp đồng”
4. Click “Thanh toán”
5. Choose payment method and process payment

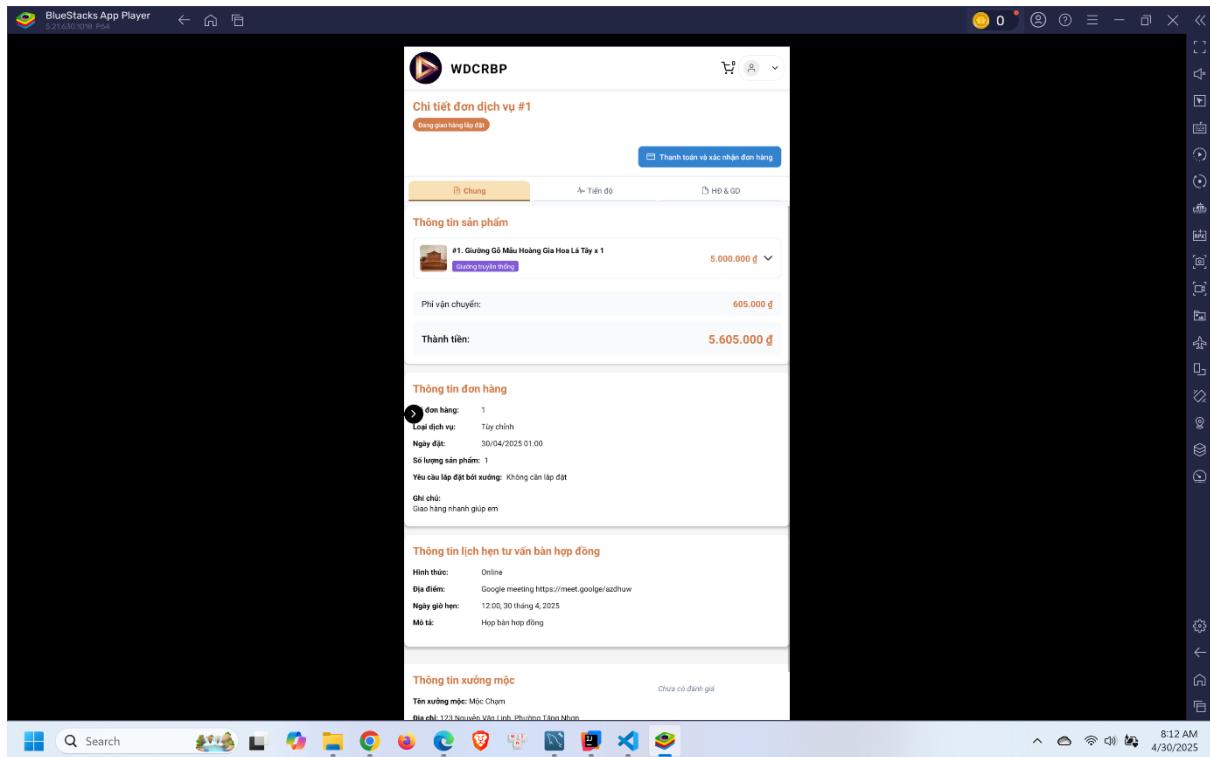
3.3.2.6 Woodworker upload the finished product image

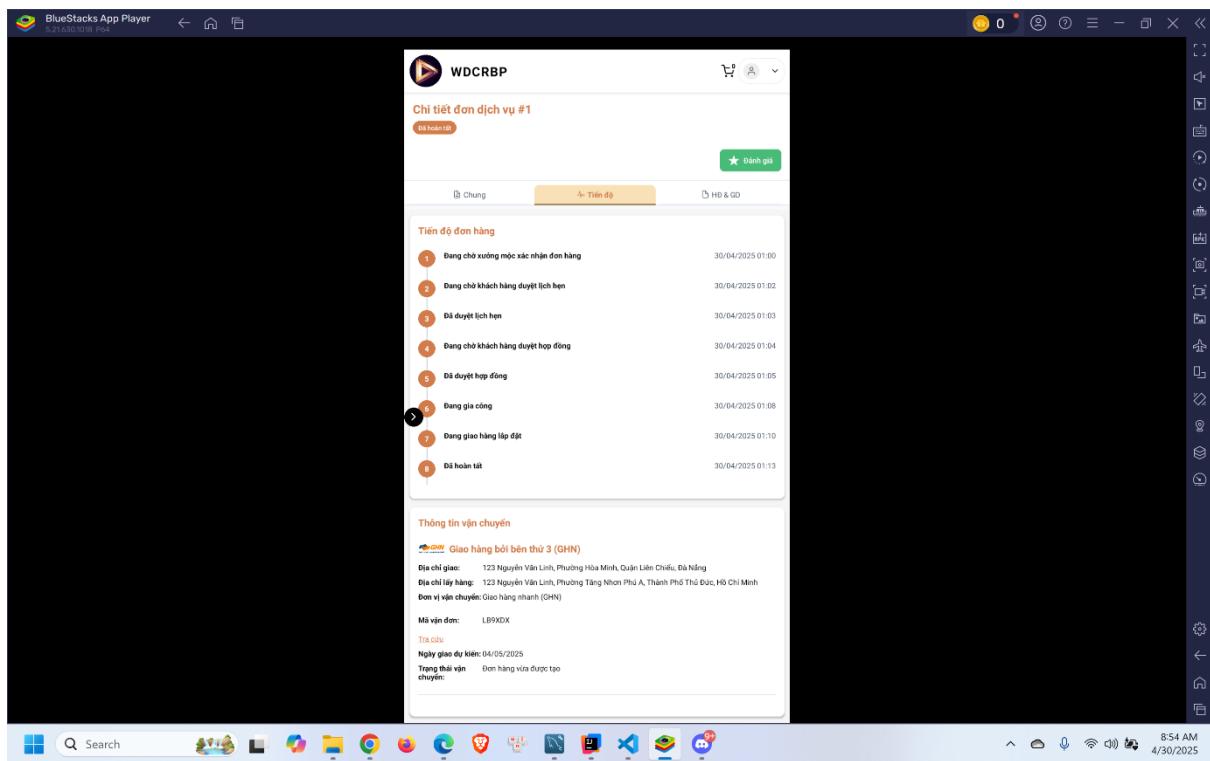
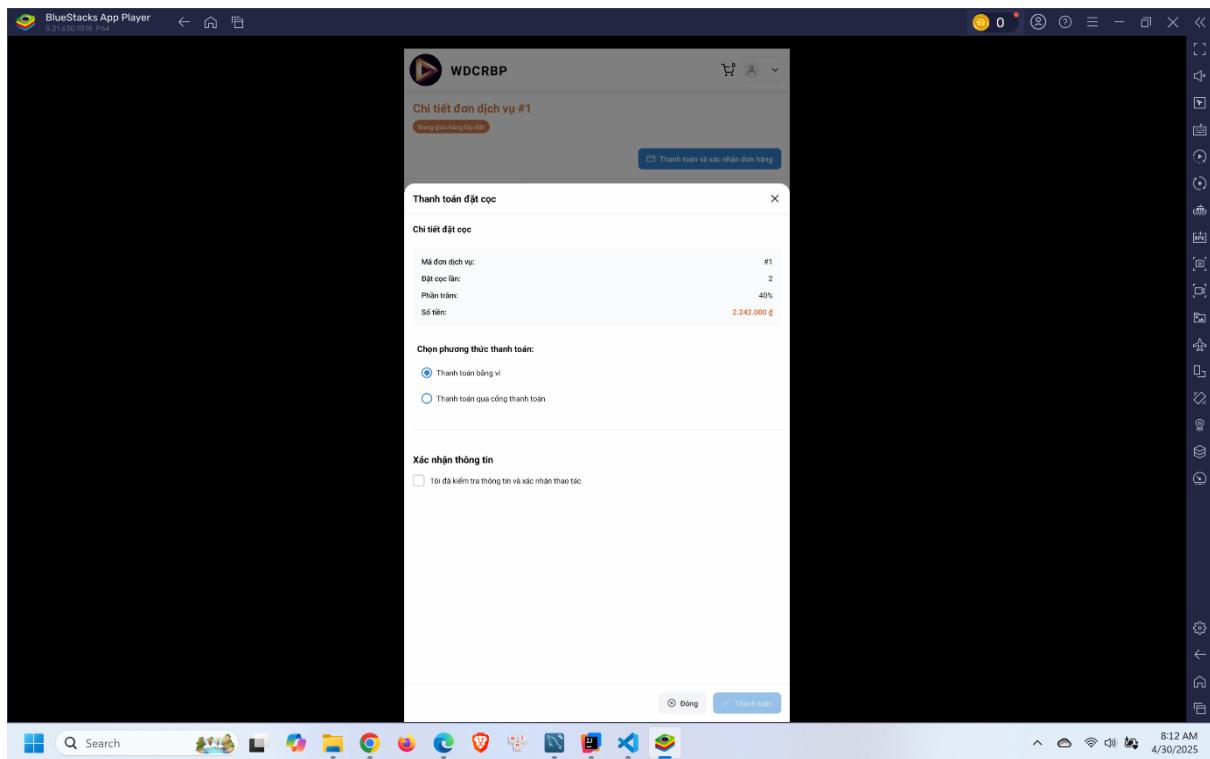




1. Click “Cập nhật ảnh hoàn thiện và giao hàng”
2. Upload images and click “Cập nhật”

3.3.2.7 Customer confirm receive product and process payment

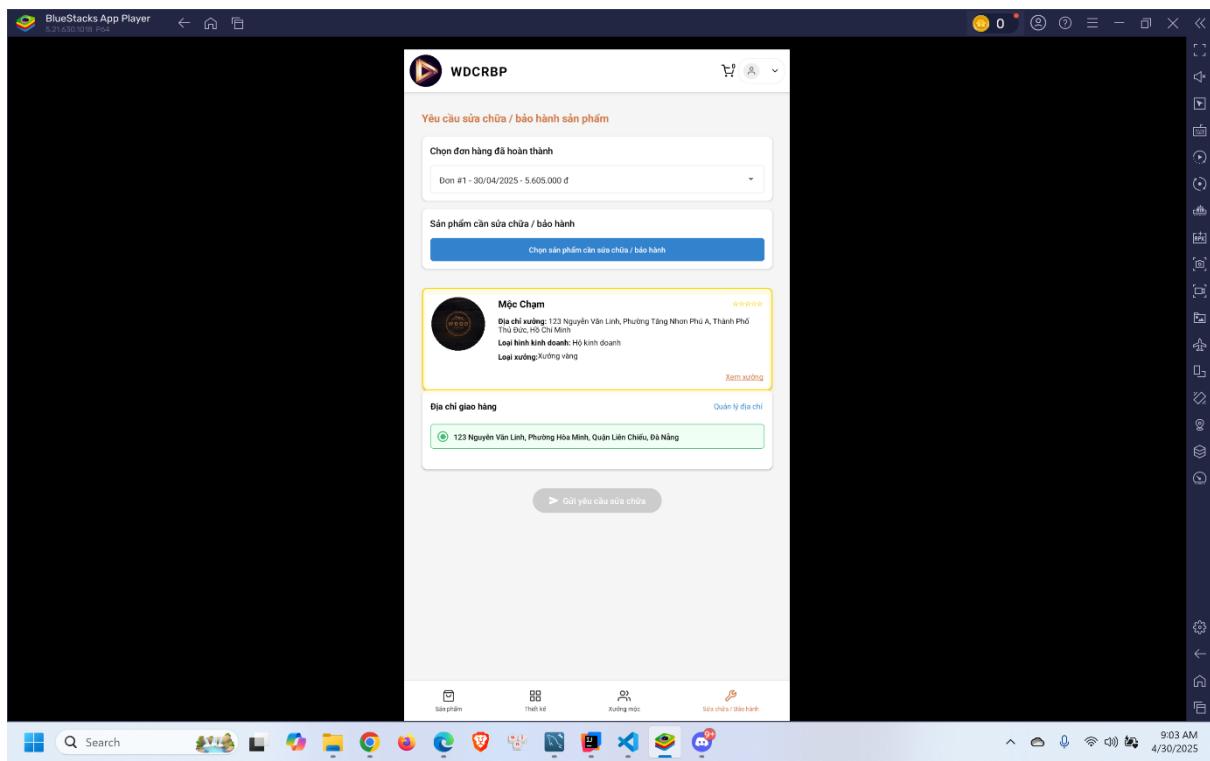
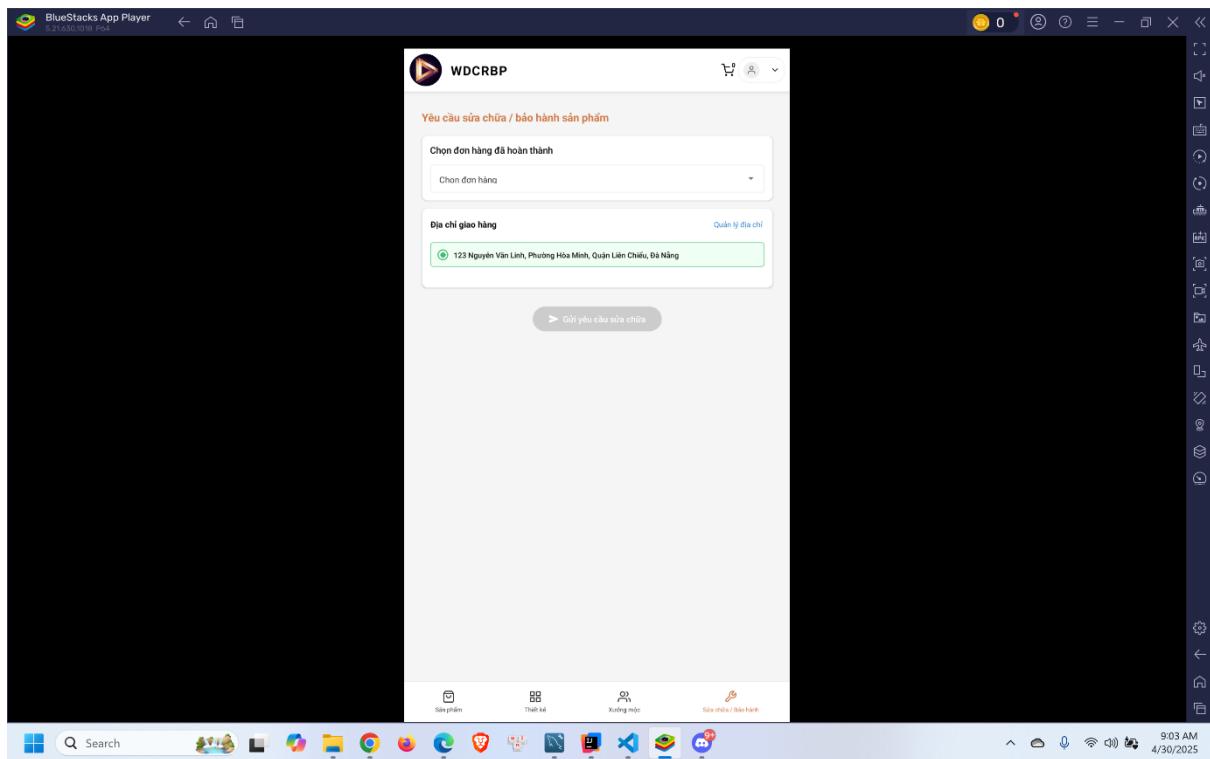


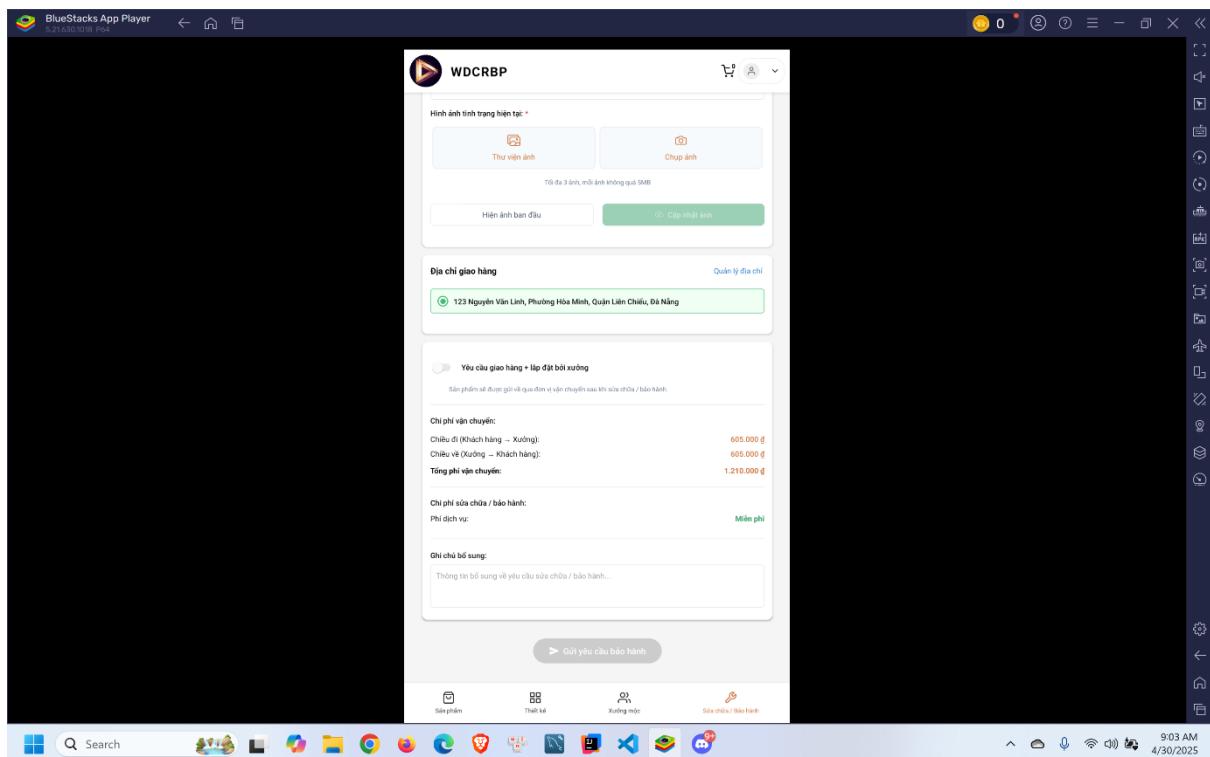
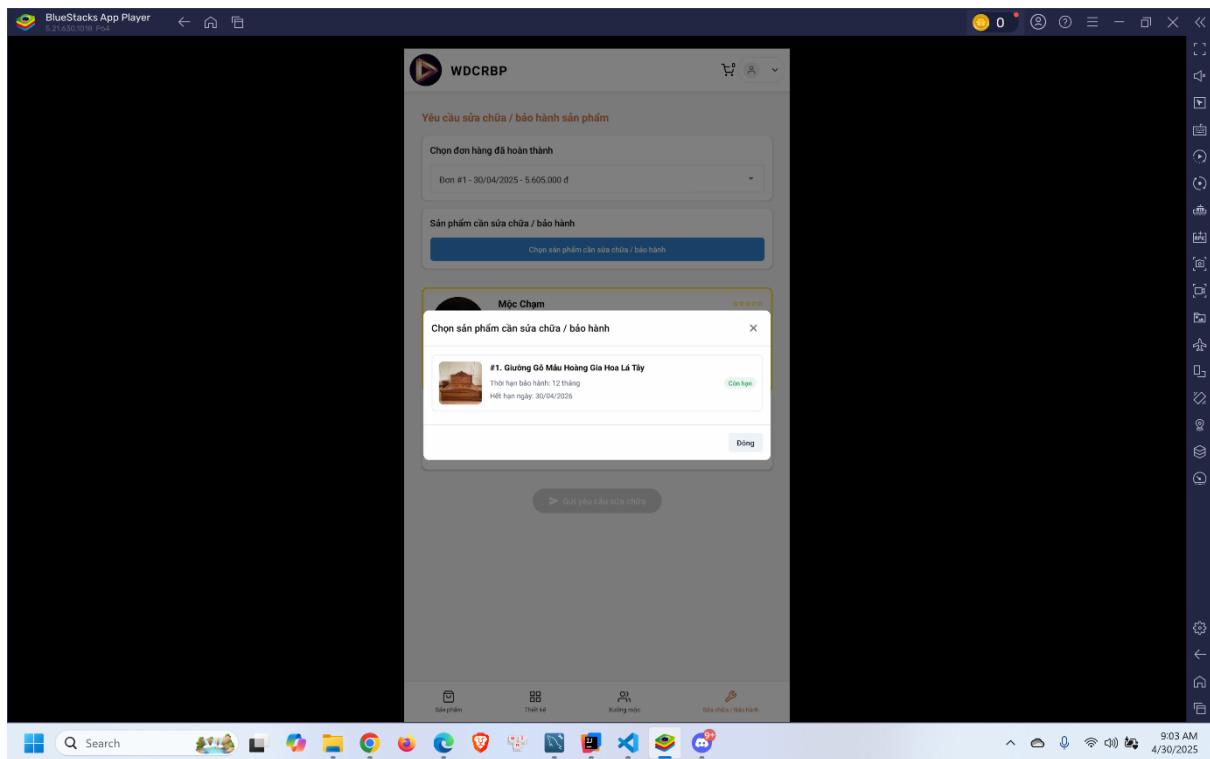


1. Click “Thanh toán và xác nhận đơn hàng”
2. Choose payment method and process payment

3.3.3 Guarantee Service Booking Feature

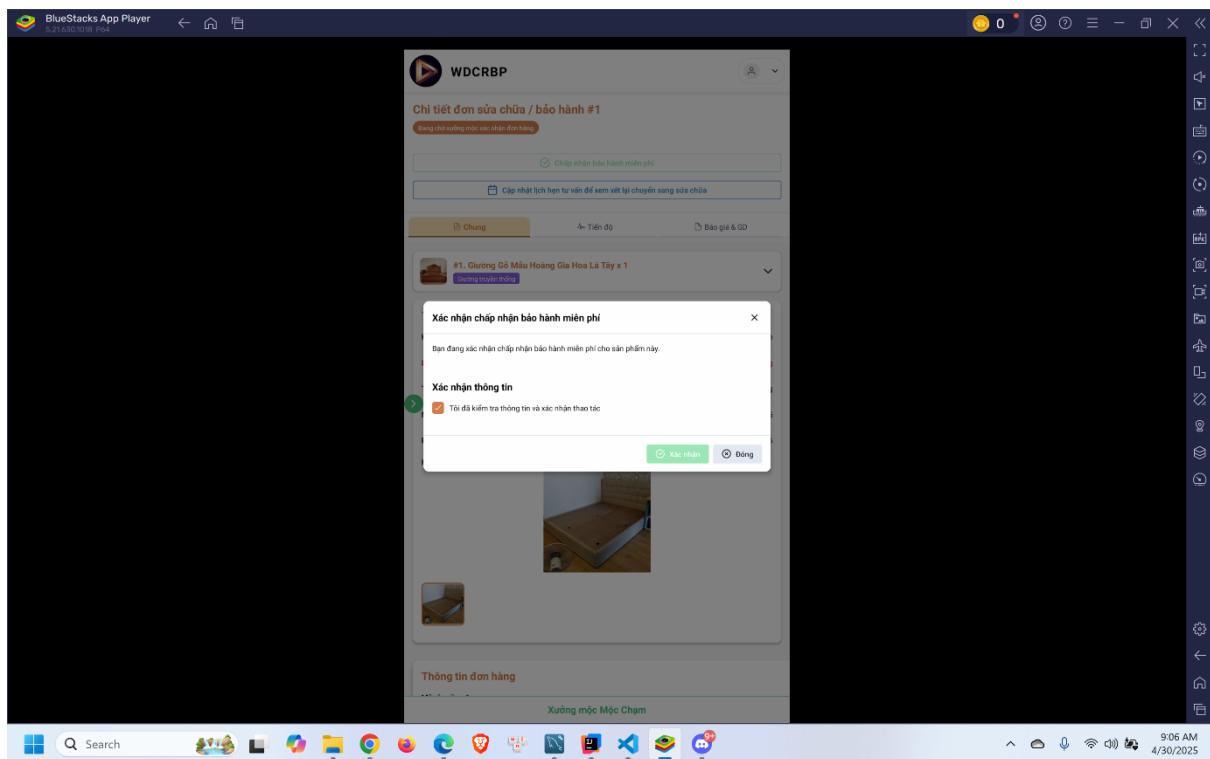
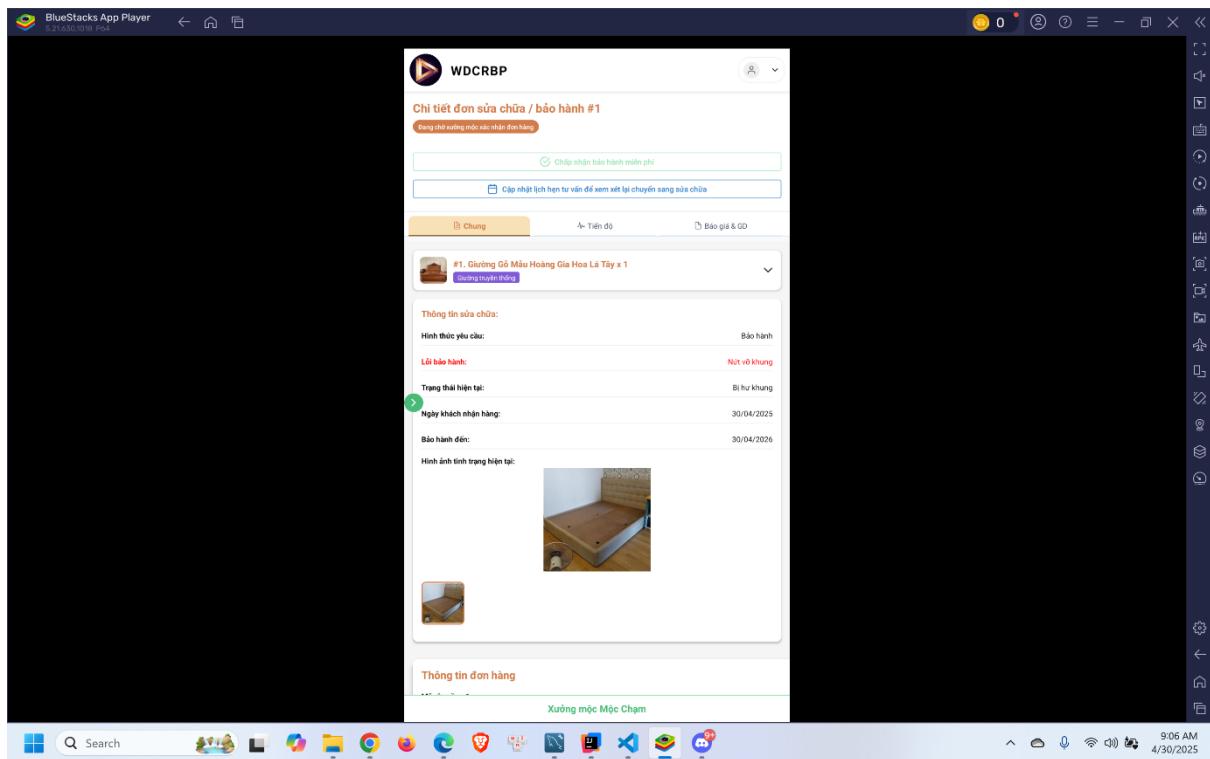
3.3.3.1 Customer request a guarantee order





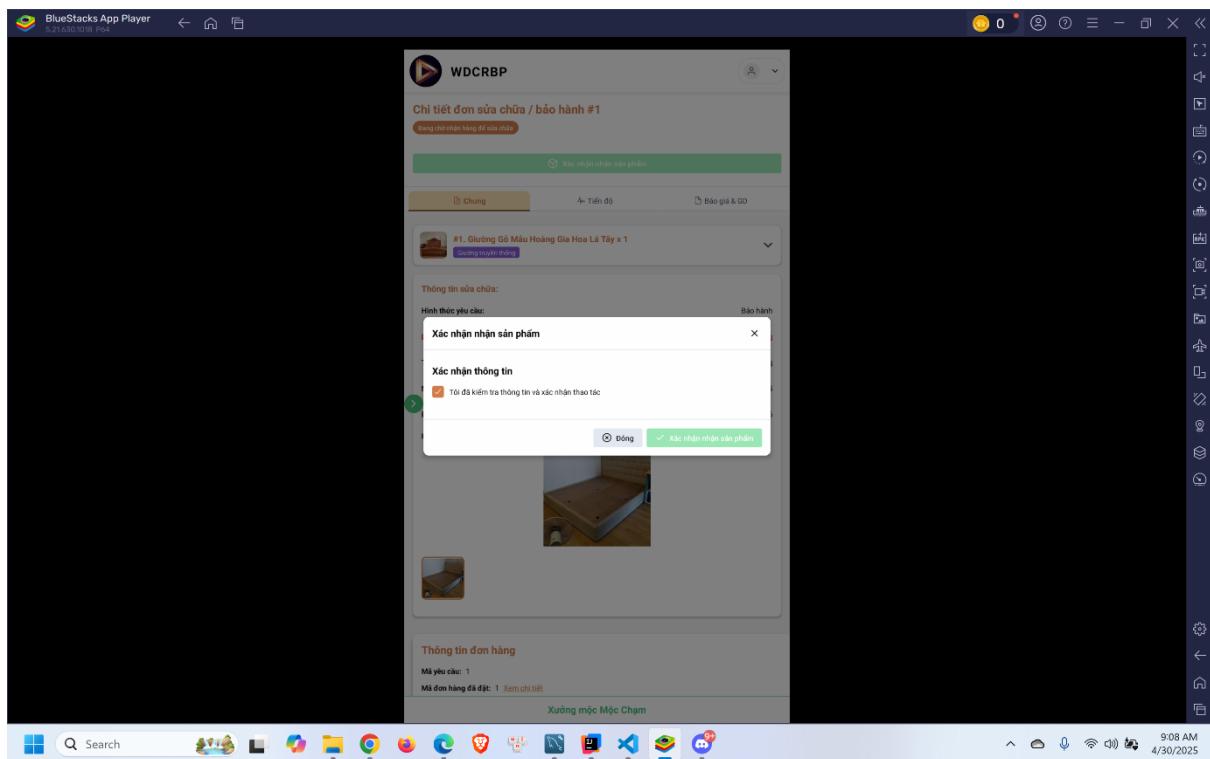
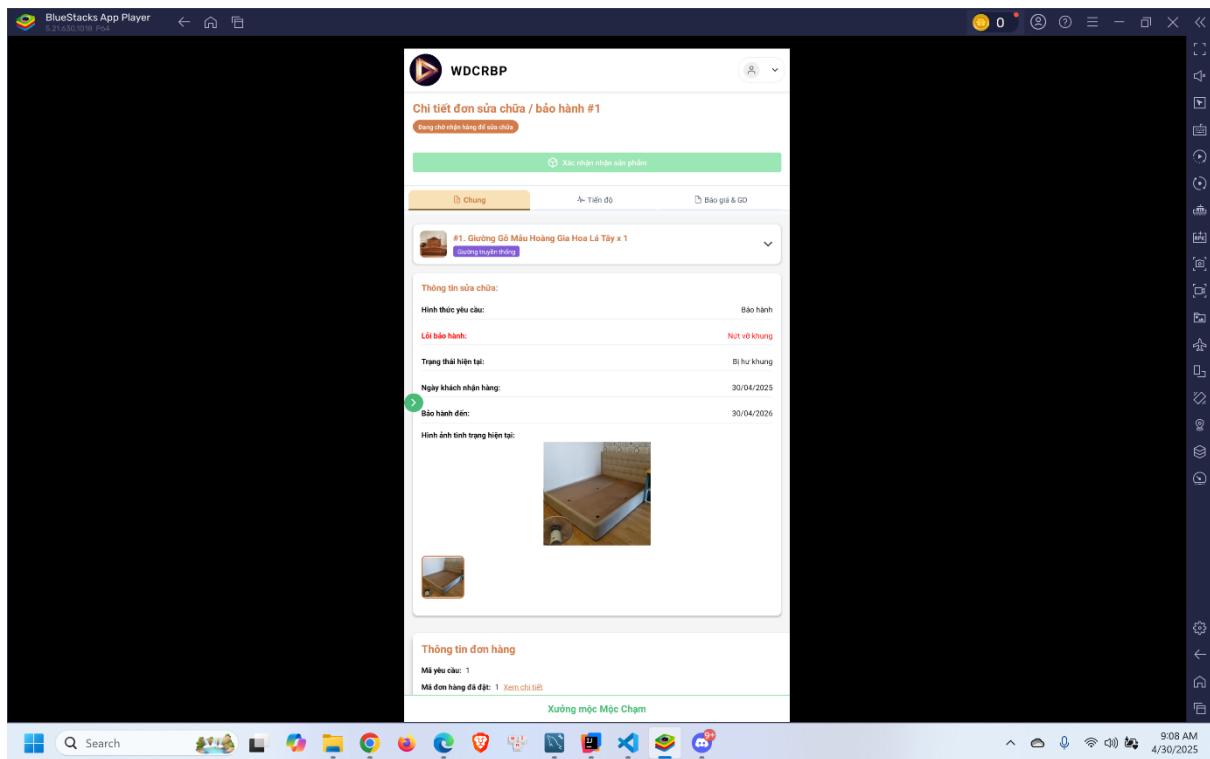
1. Click “Sửa chữa / bảo hành” in the menu
2. Select order to request
3. Select product to request
4. Fill in the form and click “Gửi yêu cầu bảo hành”

3.3.3.2 Woodworker accept the guarantee order



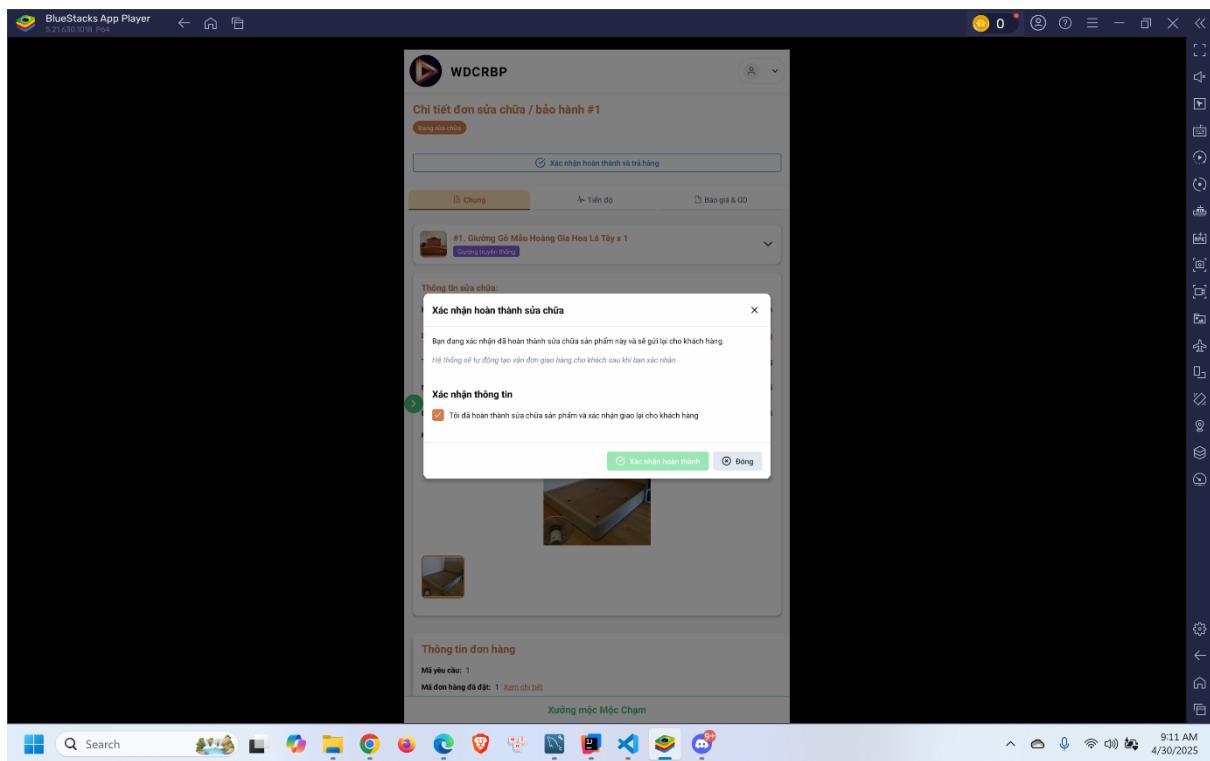
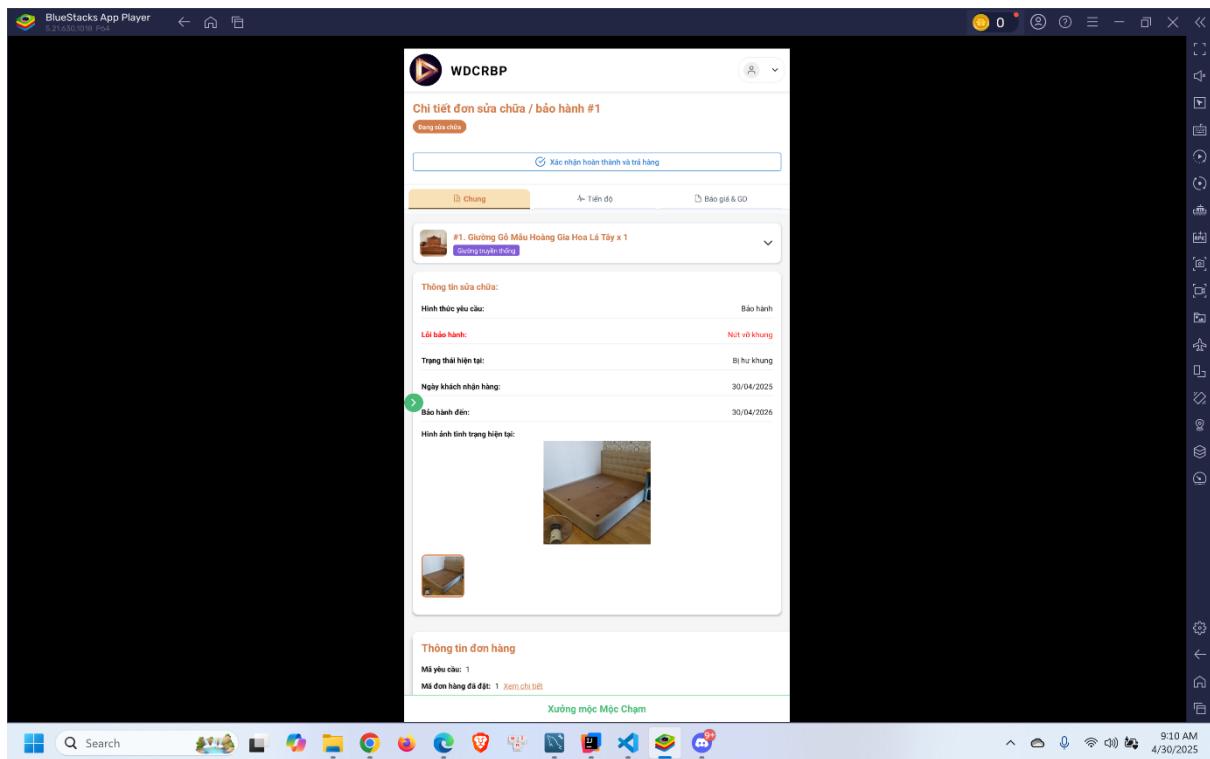
1. Click “Chấp nhận bảo hành miễn phí”
2. Click “Xác nhận”

3.3.3.3 Woodworker confirm receive product



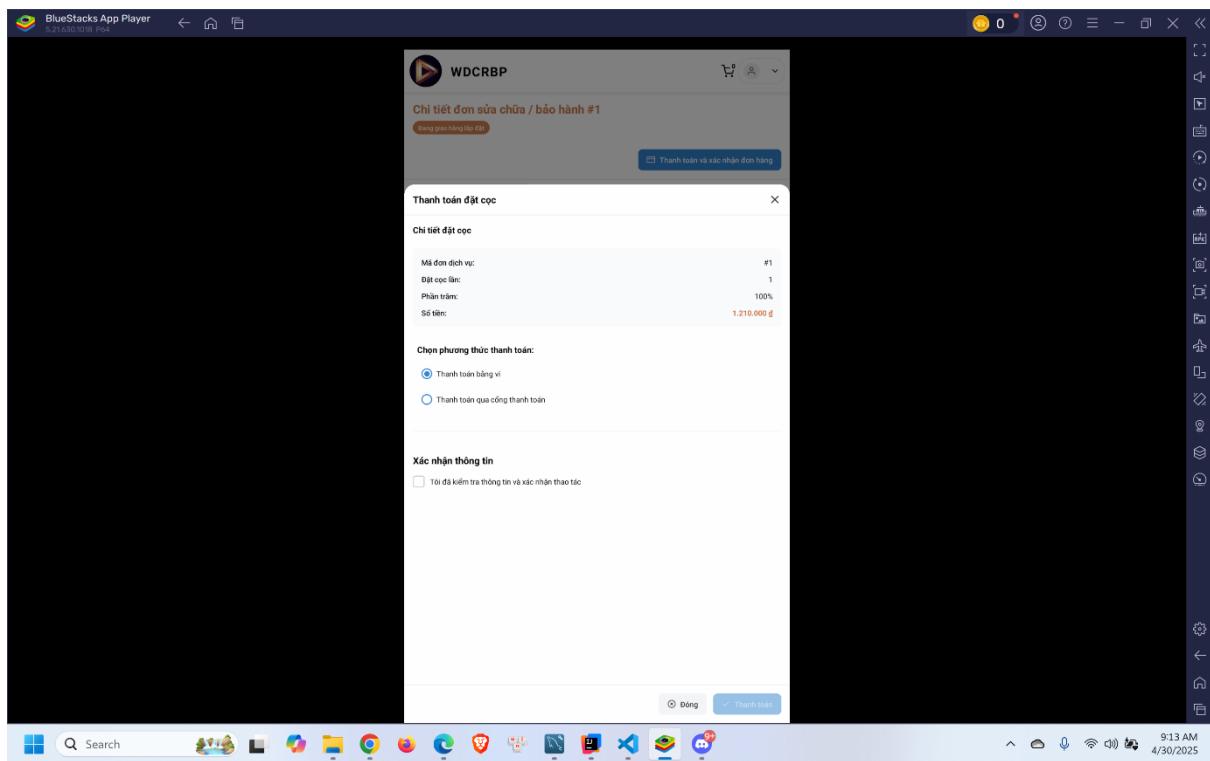
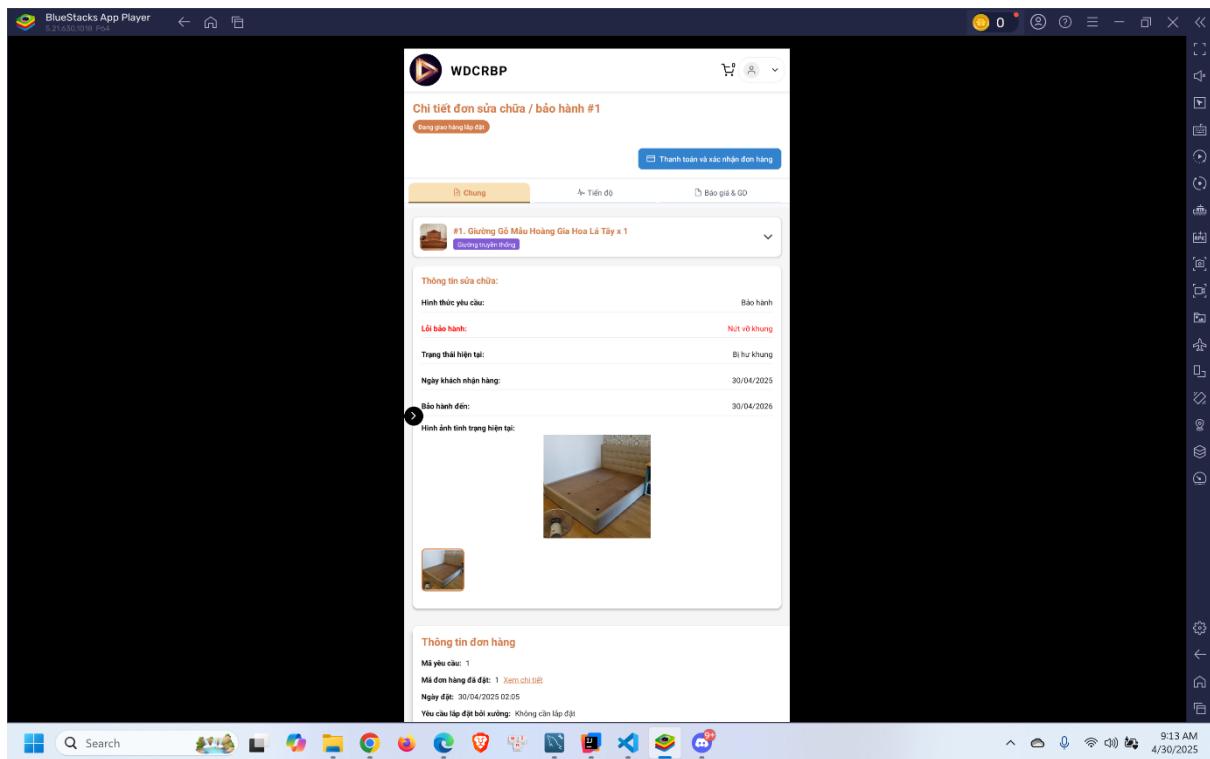
1. Click “Xác nhận nhận sản phẩm”
2. Click “Xác nhận”

3.3.3.4 Woodworker confirm repair finish



- Click “Xác nhận hoàn thành và trả hàng”
- Click “Xác nhận hoàn thành”

3.3.3.5 Customer confirm receive product and process payment



1. Click “Thanh toán và xác nhận đơn hàng”
2. Choose payment method and process payment