Appendix B: Complete Database Schema

This appendix provides the complete database schema for the Freelance Services Marketplace, including all tables, fields, data types, constraints, and relationships.

B.1 Database Overview

Database Name: **freelance_marketplace**

Character Set: **utf8mb4**

Collation: **utf8mb4_unicode_ci**

The database consists of 6 main tables with appropriate foreign key relationships to maintain referential integrity.

B.2 users Table

Purpose: Stores all user accounts for both clients and freelancers.

Field Name	Data Type	Constraints	Description	Example Value
user_id	VARCHAR(10)	PRIMARY KEY	10-digit unique identifier	1234567890
first_name	VARCHAR(50)	NOT NULL	User first name	John
last_name	VARCHAR(50)	NOT NULL	User last name	Doe
email	VARCHAR(100)	NOT NULL, UNIQUE	User email address	john@example.com
password	VARCHAR(255)	NOT NULL	Hashed password (password_hash)	$2y$10$...
phone	VARCHAR(10)	NOT NULL	10-digit phone number	5551234567
country	VARCHAR(50)	NOT NULL	User country	United States
city	VARCHAR(50)	NOT NULL	User city	New York
role	ENUM('Client', 'Freelancer')	NOT NULL	User account type	Client
status	ENUM('Active', 'Inactive')	NOT NULL, DEFAULT 'Active'	Account status	Active
registration_date	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Account creation date	2024-12-01 10:30:00

Note: *profile_photo* field (VARCHAR(255), NULLABLE) stores the file path to uploaded profile pictures.

B.3 categories Table

Purpose: Stores service categories for organizing and filtering services.

Field Name	Data Type	Constraints	Description	Example Value
category_id	INT	PRIMARY KEY, AUTO_INCREMENT	Unique category identifier	1
category_name	VARCHAR(100)	NOT NULL, UNIQUE	Category name	Web Development
Programming	VARCHAR(100)	NULLABLE	Subcategory example	Frontend Development
Design	VARCHAR(100)	NULLABLE	Subcategory example	Logo Design
Writing	VARCHAR(100)	NULLABLE	Subcategory example	Content Writing
Marketing	VARCHAR(100)	NULLABLE	Subcategory example	SEO

Predefined Categories:

• Web Development
• Graphic Design
• Writing & Translation
• Digital Marketing
• Video & Animation

B.4 services Table

Purpose: Stores all services offered by freelancers.

Field Name	Data Type	Constraints	Description	Example Value
service_id	VARCHAR(10)	PRIMARY KEY	10-digit unique identifier	9876543210
freelancer_id	VARCHAR(10)	NOT NULL, FOREIGN KEY → users(user_id)	Service owner (freelancer)	1234567890
title	VARCHAR(200)	NOT NULL	Service title	WordPress Website Development
category	VARCHAR(100)	NOT NULL	Main category	Web Development

subcategory	VARCHAR(100)	NOT NULL	Subcategory	WordPress
description	TEXT	NOT NULL	Detailed service description	I will create...
price	DECIMAL(10, 2)	NOT NULL	Service price in USD	500.00
delivery_time	INT	NOT NULL	Delivery time in days	7
revisions_included	INT	NOT NULL	Number of revisions (999=unlimited)	3
image_1	VARCHAR(255)	NOT NULL	Primary service image path	/uploads/services/...
image_2	VARCHAR(255)	NULLABLE	Second service image path	/uploads/services/...
image_3	VARCHAR(255)	NULLABLE	Third service image path	/uploads/services/...
status	ENUM('Active', 'Inactive')	NOT NULL, DEFAULT 'Active'	Service availability	Active
featured_status	ENUM('Yes', 'No')	NOT NULL, DEFAULT 'No'	Featured on homepage	No
created_date	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Service creation date	2024-12-01 15:00:00

Foreign Key Constraints:

• freelancer_id references users(user_id) ON DELETE CASCADE

B.5 orders Table

Purpose: Stores all order transactions between clients and freelancers.

Field Name	Data Type	Constraints	Description	Example Value
order_id	VARCHAR(10)	PRIMARY KEY	10-digit unique identifier	1122334455
client_id	VARCHAR(10)	NOT NULL, FOREIGN KEY → users(user_id)	Client who placed order	1234567890
freelancer_id	VARCHAR(10)	NOT NULL, FOREIGN KEY → users(user_id)	Freelancer fulfilling order	9876543210

service_id	VARCHAR(10)	NOT NULL, FOREIGN KEY → services(service_id)	Service being ordered	5555555555
service_title	VARCHAR(200)	NOT NULL	Service title (snapshot)	WordPress Website
price	DECIMAL(10,2)	NOT NULL	Order price (snapshot)	500.00
delivery_time	INT	NOT NULL	Delivery days (snapshot)	7
revisions_included	INT	NOT NULL	Revisions allowed (snapshot)	3
requirements	TEXT	NOT NULL	Client project requirements	I need a website for...
deliverable_notes	TEXT	NULLABLE	Freelancer delivery message	Here is your completed...
status	ENUM(...)	NOT NULL, DEFAULT 'Pending'	Order status (see below)	In Progress
payment_method	VARCHAR(50)	NOT NULL	Payment method used	Credit Card
order_date	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Order placement date	2024-12-01 16:00:00
expected_delivery	DATE	NOT NULL	Expected delivery date	2024-12-08
completion_date	TIMESTAMP	NULLABLE	Actual completion date	2024-12-07 14:30:00

Order Status Values:

• Pending - Order placed, awaiting freelancer start

• In Progress - Freelancer working on order

• Delivered - Freelancer submitted deliverable

• Completed - Client accepted delivery

• Revision Requested - Client requested changes

• Cancelled - Order cancelled

Foreign Key Constraints:

- client_id references users(user_id) ON DELETE CASCADE
- freelancer_id references users(user_id) ON DELETE CASCADE
- service_id references services(service_id) ON DELETE RESTRICT

B.6 revision_requests Table

Purpose: Tracks revision requests made by clients on delivered orders.

Field Name	Data Type	Constraints	Description	Example Value
revision_id	INT	PRIMARY KEY, AUTO_INCREMENT	Unique revision request ID	1
order_id	VARCHAR(10)	NOT NULL, FOREIGN KEY → orders(order_id)	Associated order	1122334455
revision_notes	TEXT	NOT NULL	Client revision instructions	Please change the header...
revision_file	VARCHAR(255)	NULLABLE	Optional reference file	/uploads/revisions/...
request_status	ENUM(...)	NOT NULL, DEFAULT 'Pending'	Revision status	Accepted
request_date	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Request submission date	2024-12-08 10:00:00
response_date	TIMESTAMP	NULLABLE	Freelancer response date	2024-12-08 11:00:00
freelancer_response	TEXT	NULLABLE	Freelancer response message	I will make these changes

Revision Status Values:

- Pending - Awaiting freelancer response
- Accepted - Freelancer agreed to make changes
- Rejected - Freelancer declined (with explanation)

Foreign Key Constraints:

- order_id references orders(order_id) ON DELETE CASCADE

B.6 file_attachments Table (NEW)

Purpose: Stores multiple file attachments for orders (requirements, deliverables, revisions)

Field Name	Data Type	Constraints	Description	Example Value
file_id	INT	PRIMARY KEY, AUTO_INCREMENT	Unique file identifier	1
order_id	VARCHAR(10)	NOT NULL, FOREIGN KEY → orders(order_id)	Associated order	1234567890
file_path	VARCHAR(255)	NOT NULL	Full path to file on server	/uploads/orders/1234567890/requirements/req_001.pdf
original_filename	VARCHAR(255)	NOT NULL	Original filename from upload	specifications.pdf
file_size	INT	NOT NULL	File size in bytes	2458000
file_type	ENUM	NOT NULL	Type of file	requirement
upload_timestamp	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	When file was uploaded	2024-12-01 10:00:00

File Type Values:

- **requirement** - Client requirement files (uploaded during checkout)
- **deliverable** - Freelancer deliverable files (uploaded when completing order)
- **revision** - Revision files (uploaded when responding to revision requests)

Foreign Key Constraints:

- order_id references orders(order_id) ON DELETE CASCADE

Notes:

- Supports **multiple files per order** (up to 3 requirement files, multiple deliverables)
- Files automatically deleted when order is deleted (CASCADE)
- Original filename stored for display/download

- File size used for validation and display

B.8 Entity Relationship Diagram

The following describes the relationships between tables:

- users (1) → (Many) services: One freelancer can offer multiple services
- users (1) → (Many) orders (as client): One client can place multiple orders
- users (1) → (Many) orders (as freelancer): One freelancer can receive multiple orders
- services (1) → (Many) orders: One service can be ordered multiple times
- orders (1) → (Many) revision_requests: One order can have multiple revision requests
- orders (1) → (Many) file_attachments: One order can have multiple file attachment

B.9 Sample Database Creation Script

Below is a sample SQL script to create the database schema:

```
CREATE DATABASE IF NOT EXISTS freelance_marketplace
CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;

USE freelance_marketplace;

-- Users Table
CREATE TABLE users (
    user_id VARCHAR(10) PRIMARY KEY,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    email VARCHAR(100) NOT NULL UNIQUE,
    password VARCHAR(255) NOT NULL,
    phone VARCHAR(10) NOT NULL,
    country VARCHAR(50) NOT NULL,
    city VARCHAR(50) NOT NULL,
    role ENUM('Client', 'Freelancer') NOT NULL,
    status ENUM('Active', 'Inactive') NOT NULL DEFAULT 'Active',
    profile_photo VARCHAR(255),
    registration_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Services Table
CREATE TABLE services (
    service_id VARCHAR(10) PRIMARY KEY,
    freelancer_id VARCHAR(10) NOT NULL,
    title VARCHAR(200) NOT NULL,
    category VARCHAR(100) NOT NULL,
    subcategory VARCHAR(100) NOT NULL,
    description TEXT NOT NULL,
    price DECIMAL(10,2) NOT NULL,
    delivery_time INT NOT NULL,
    revisions_included INT NOT NULL,
    image_1 VARCHAR(255) NOT NULL,
    image_2 VARCHAR(255),
    image_3 VARCHAR(255),
    status ENUM('Active', 'Inactive') NOT NULL DEFAULT 'Active',
```

```sql
    featured_status ENUM('Yes', 'No') NOT NULL DEFAULT 'No',
    created_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (freelancer_id) REFERENCES users(user_id) ON DELETE CASCADE
);

-- Orders Table
CREATE TABLE orders (
    order_id VARCHAR(10) PRIMARY KEY,
    client_id VARCHAR(10) NOT NULL,
    freelancer_id VARCHAR(10) NOT NULL,
    service_id VARCHAR(10) NOT NULL,
    service_title VARCHAR(200) NOT NULL,
    price DECIMAL(10,2) NOT NULL,
    delivery_time INT NOT NULL,
    revisions_included INT NOT NULL,
    requirements TEXT NOT NULL,
    deliverable_notes TEXT,
    status ENUM('Pending', 'In Progress', 'Delivered', 'Completed',
                'Revision Requested', 'Cancelled') NOT NULL DEFAULT 'Pending',
    payment_method VARCHAR(50) NOT NULL,
    order_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    expected_delivery DATE NOT NULL,
    completion_date TIMESTAMP,
    FOREIGN KEY (client_id) REFERENCES users(user_id) ON DELETE CASCADE,
    FOREIGN KEY (freelancer_id) REFERENCES users(user_id) ON DELETE CASCADE,
    FOREIGN KEY (service_id) REFERENCES services(service_id) ON DELETE RESTRICT
);

-- Revision Requests Table
CREATE TABLE revision_requests (
    revision_id INT AUTO_INCREMENT PRIMARY KEY,
    order_id VARCHAR(10) NOT NULL,
    revision_notes TEXT NOT NULL,
    revision_file VARCHAR(255),
    request_status ENUM('Pending', 'Accepted', 'Rejected') NOT NULL DEFAULT
'Pending',
    request_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    response_date TIMESTAMP,
    freelancer_response TEXT,
    FOREIGN KEY (order_id) REFERENCES orders(order_id) ON DELETE CASCADE
);

CREATE TABLE file_attachments (
    file_id INT AUTO_INCREMENT PRIMARY KEY,
    order_id VARCHAR(10) NOT NULL,
    file_path VARCHAR(255) NOT NULL,
    original_filename VARCHAR(255) NOT NULL,
    file_size INT NOT NULL,
    file_type ENUM('requirement', 'deliverable', 'revision') NOT NULL,
    upload_timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    FOREIGN KEY (order_id) REFERENCES orders(order_id) ON DELETE CASCADE
);
```