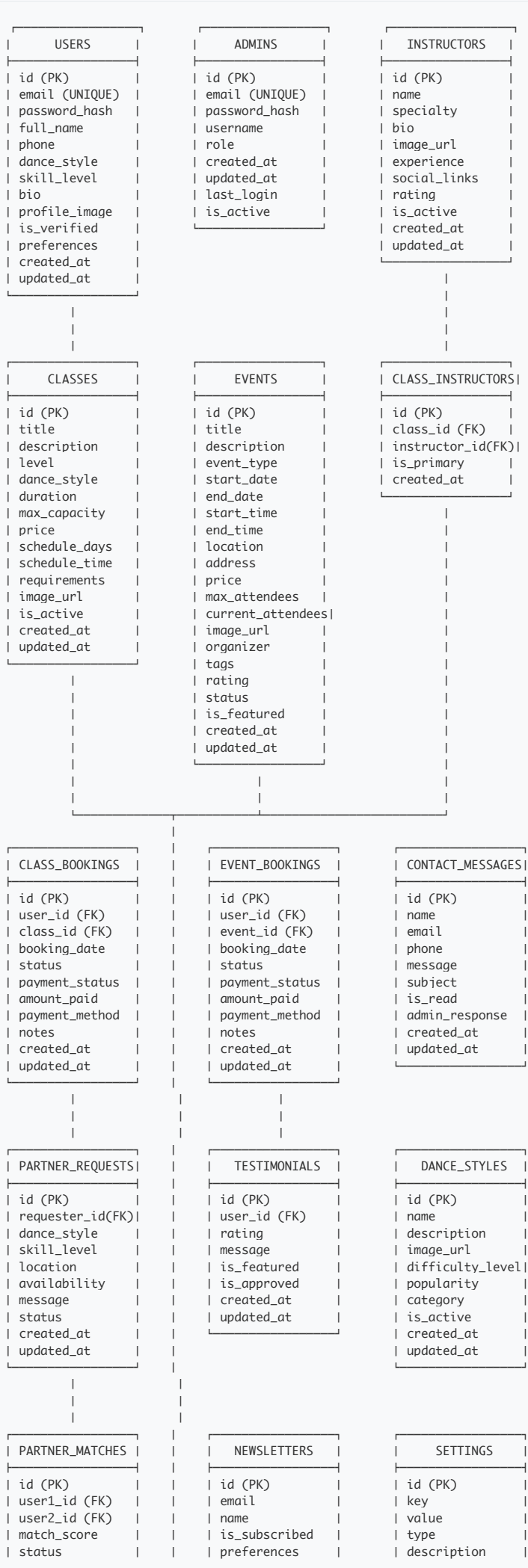
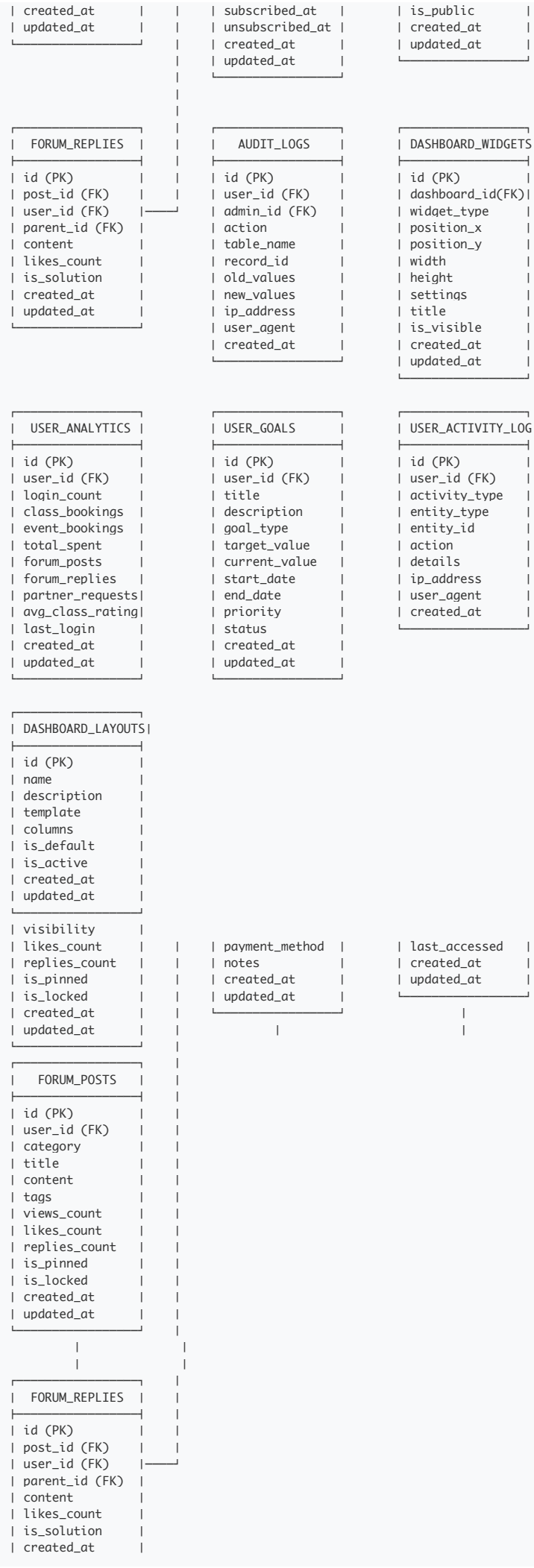
DanceLink - Entity Relationship Diagram (ERD)

Database Schema Design

Core Entities and Relationships


USERS
id (PK)
email (UNIQUE)
password_hash
full_name
phone
dance_style
skill_level
bio
profile_image
is_verified
preferences
created_at
updated_at
ADMINS
id (PK)
email (UNIQUE)
password_hash
username
role
created_at
updated_at
last_login
is_active
INSTRUCTORS
id (PK)
name
specialty
bio
image_url
experience
social_links
rating
is_active
created_at
updated_at
CLASSES
id (PK)
title
description
level
dance_style
duration
max_capacity
price
schedule_days
schedule_time
requirements
image_url
is_active
created_at
updated_at
EVENTS
id (PK)
title
description
event_type
start_date
end_date
start_time
end_time
location
address
price
max_attendees
current_attendees
image_url
organizer
tags
rating
status
is_featured
created_at
updated_at
CLASS_INSTRUCTORS
id (PK)
class_id (FK)
instructor_id(FK)
is_primary
created_at
CLASS_BOOKINGS
id (PK)
user_id (FK)
class_id (FK)
booking_date
status
payment_status
amount_paid
payment_method
notes
created_at
updated_at
EVENT_BOOKINGS
id (PK)
user_id (FK)
event_id (FK)
booking_date
status
payment_status
amount_paid
payment_method
notes
created_at
updated_at
CONTACT_MESSAGES
id (PK)
name
email
phone
message
subject
is_read
admin_response
created_at
updated_at
PARTNER_REQUESTS
id (PK)
requester_id(FK)
dance_style
skill_level
location
availability
message
status
created_at
updated_at
TESTIMONIALS
id (PK)
user_id (FK)
rating
message
is_featured
is_approved
created_at
updated_at
DANCE_STYLES
id (PK)
name
description
image_url
difficulty_level
popularity
category
is_active
created_at
updated_at
PARTNER_MATCHES
id (PK)
user1_id (FK)
user2_id (FK)
match_score
status
NEWSLETTERS
id (PK)
email
name
is_subscribed
preferences
SETTINGS
id (PK)
key
value
type
description


created_at
updated_at
subscribed_at
unsubscribed_at
created_at
updated_at
is_public
created_at
updated_at
FORUM_REPLIES
id (PK)
post_id (FK)
user_id (FK)
parent_id (FK)
content
likes_count
is_solution
created_at
updated_at
AUDIT_LOGS
id (PK)
user_id (FK)
admin_id (FK)
action
table_name
record_id
old_values
new_values
ip_address
user_agent
created_at
DASHBOARD_WIDGETS
id (PK)
dashboard_id(FK)
widget_type
position_x
position_y
width
height
settings
title
is_visible
created_at
updated_at
USER_ANALYTICS
id (PK)
user_id (FK)
login_count
class_bookings
event_bookings
total_spent
forum_posts
forum_replies
partner_requests
avg_class_rating
last_login
created_at
updated_at
USER_GOALS
id (PK)
user_id (FK)
title
description
goal_type
target_value
current_value
start_date
end_date
priority
status
created_at
updated_at
USER_ACTIVITY_LOG
id (PK)
user_id (FK)
activity_type
entity_type
entity_id
action
details
ip_address
user_agent
created_at
DASHBOARD_LAYOUTS
id (PK)
name
description
template
columns
is_default
is_active
created_at
updated_at
visibility
likes_count
replies_count
is_pinned
is_locked
created_at
updated_at
payment_method
notes
created_at
updated_at
last_accessed
created_at
updated_at
FORUM_POSTS
id (PK)
user_id (FK)
category
title
content
tags
views_count
likes_count
replies_count
is_pinned
is_locked
created_at
updated_at
FORUM_REPLIES
id (PK)
post_id (FK)
user_id (FK)
parent_id (FK)
content
likes_count
is_solution
created_at

```
| updated_at      |
└─────────────────┘
```

Relationships

One-to-Many Relationships

- **USERS** → **CLASS_BOOKINGS** (1:N)
- **USERS** → **EVENT_BOOKINGS** (1:N)
- **USERS** → **PARTNER_REQUESTS** (1:N)
- **USERS** → **TESTIMONIALS** (1:N)
- **USERS** → **NOTIFICATIONS** (1:N)
- **USERS** → **FORUM_POSTS** (1:N)
- **USERS** → **FORUM_REPLIES** (1:N)
- **USERS** → **AUDIT_LOGS** (1:N)
- **USERS** → **USER_DASHBOARDS** (1:N)
- **USERS** → **USER_ANALYTICS** (1:1)
- **USERS** → **USER_GOALS** (1:N)
- **USERS** → **USER_ACTIVITY_LOG** (1:N)
- **USER_DASHBOARDS** → **DASHBOARD_WIDGETS** (1:N)
- **CLASSES** → **CLASS_BOOKINGS** (1:N)
- **EVENTS** → **EVENT_BOOKINGS** (1:N)
- **FORUM_POSTS** → **FORUM_REPLIES** (1:N)
- **FORUM_REPLIES** → **FORUM_REPLIES** (1:N - self-referencing for nested replies)

Many-to-Many Relationships

- **CLASSES** ↔ **INSTRUCTORS** (through CLASS_INSTRUCTORS)
- **USERS** ↔ **USERS** (through PARTNER_MATCHES for partner matching)
- **USER_DASHBOARDS** ↔ **DASHBOARD_LAYOUTS** (through dashboard_id and layout configuration)

Key Constraints

Primary Keys (PK)

- All tables have an auto-incrementing `id` as primary key

Foreign Keys (FK)

- `user_id` references USERS(id)
- `admin_id` references ADMINS(id)
- `class_id` references CLASSES(id)
- `event_id` references EVENTS(id)
- `instructor_id` references INSTRUCTORS(id)
- `post_id` references FORUM_POSTS(id)
- `parent_id` references FORUM_REPLIES(id)
- `dashboard_id` references USER_DASHBOARDS(id)

Unique Constraints

- USERS.email
- ADMINS.email
- ADMINS.username

Indexes (for performance)

- email fields (USERS, ADMINS, NEWSLETTERS)
- foreign key fields
- created_at/updated_at timestamps
- status fields
- search fields (name, title, category)

Data Types and Constraints

Common Field Types

- **id**: BIGINT AUTO_INCREMENT PRIMARY KEY

- **email**: VARCHAR(255) UNIQUE NOT NULL
- **password_hash**: VARCHAR(255) NOT NULL
- **created_at/updated_at**: TIMESTAMP DEFAULT CURRENT_TIMESTAMP
- **is_active/is_featured/is_read**: BOOLEAN DEFAULT TRUE/FALSE
- **status**: ENUM('active', 'inactive', 'pending', 'cancelled', 'completed')
- **rating**: DECIMAL(3,2) DEFAULT 0.00 (0.00 to 5.00)
- **price/amount_paid**: DECIMAL(10,2)
- **JSON fields**: preferences, social_links, tags (MySQL JSON type)

Business Rules

1. Users can book multiple classes and events
2. Instructors can teach multiple classes
3. Classes can have multiple instructors (primary and assistant)
4. Events have capacity limits
5. Partner matching is mutual (both users must agree)
6. Forum replies can be nested (threaded discussions)
7. Admin actions are logged for audit trail
8. Soft deletes for important data (is_active flag)
9. Each user can have multiple dashboards but only one default
10. Dashboard widgets are configurable and can be resized/repositioned
11. User analytics are automatically calculated based on user activity
12. User goals track progress toward specific dance achievements
13. Activity logs track all user interactions for analytics purposes

Security Considerations

- Password hashing (bcrypt)
- JWT tokens for authentication
- Role-based access control (RBAC)
- Input validation and sanitization
- Audit logging for admin actions
- Rate limiting on API endpoints