

项目名称

项目\
项目\
项目 Spring Boot \
**XXX XXX XXXXXXXXX \
**XXX XXX XXX \
**2026 X X X

二〇二六

“项目”指的是一个综合性的家庭旅馆推荐系统，“项目 + 版本”指的是该系统的不同迭代版本。Vue 3 + Vite + Element Plus 构成了前端，Spring Boot、MyBatis-Plus、JWT 和 MySQL 构成了后端。该项目旨在解决短期租赁平台上信息过载的问题，通过结合协作过滤和内容相似性推荐策略，为用户提供个性化推荐。

项目\Spring Boot\MySQL\

Abstract

This report presents the design and implementation of a homestay recommendation system. The system targets the information overload in short-term rental platforms and uses a hybrid recommendation strategy combining collaborative filtering and content similarity. The frontend is implemented with Vue 3 and Element Plus, while the backend uses Spring Boot, MyBatis-Plus, JWT authentication, and MySQL storage. The report follows the database course project template, covering requirement analysis, conceptual and logical database design, physical implementation, and database operations, and further describes the overall architecture and recommender logic. The project provides a practical reference for similar personalized recommendation systems.

Keywords: Homestay recommendation; Collaborative filtering; Content-based similarity; Spring Boot; MySQL

100 / 100

| □□ | □□ |

| --- | --- |

| | |

| 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 |

| 0000 | 2022105420004 XXXXXXXXXXXX |

| 1 E-R 3 MySQL 5 |

1

1. □□□□＼
 2. □□□□□＼
 3. □□□□□＼
 4. □□□□□□□＼
 5. □□□□＼
 6. □□□□＼
 7. □□□□□□＼
 8. □□□□＼
 9. □□□＼
 10. □□□

1 □□□□

1.1 မြန်မာစာ

၁။ “၂။ + ၃။” မြန်မာစာတွင် ၂။ နှင့် ၃။ အတွက် “၂။။ + ၃။။” မြန်မာစာတွင် ၂။။ နှင့် ၃။။ အတွက်

၂။။ နှင့် ၃။။ အတွက် “၂။။။ + ၃။။။” မြန်မာစာတွင် ၂။။။ နှင့် ၃။။။ အတွက်

1.2 မြန်မာစာ

၂။။။ နှင့် ၃။။။ အတွက်

- ၂။။။။
- ၃။။။။
- ၂။။။။။
- ၃။။။။။
- ၂။။။။။။
- ၃။။။။။။

1.3 မြန်မာစာ

၁။ **မြန်မာစာ** မြန်မာစာတွင် ၁။ နှင့် ၁။။ အတွက်

၂။ **၂။** မြန်မာစာတွင် ၂။ နှင့် ၂။။ အတွက်

၃။ **၃။** မြန်မာစာတွင် ၃။ နှင့် ၃။။ အတွက်

1.4 မြန်မာစာ

- **၂။။။** မြန်မာစာတွင် ၂။။။ နှင့် ၂။။။။ အတွက်
- **၃။။။** မြန်မာစာတွင် ၃။။။ နှင့် ၃။။။။ အတွက်

- **[REDACTED]** [REDACTED]
 - **[REDACTED]** [REDACTED]
 - **[REDACTED]** [REDACTED]
 - **[REDACTED]** [REDACTED]

1.5

- **[REDACTED]**
 - **[REDACTED]** JWT [REDACTED]
 - **[REDACTED]**
 - **[REDACTED]**
 - **[REDACTED]**

1.6

1.7

- **[REDACTED]** [REDACTED]
 - **[REDACTED]** [REDACTED]
 - **[REDACTED]** [REDACTED] Token [REDACTED]
 - **[REDACTED]** [REDACTED]

1.8

[REDACTED] [REDACTED]
[REDACTED]

[REDACTED] [REDACTED]
[REDACTED]

DAU
DAU
DAU

MAU
MAU
MAU

UV
UV
UV

ARPU
ARPU
ARPU

ROI
ROI
ROI

CPA
CPA
CPA

1.9 សេវាអនុញ្ញាត

សេវាអនុញ្ញាត DAU/MAU

សេវាអនុញ្ញាត MAU

សេវាអនុញ្ញាត UV

សេវាអនុញ្ញាត ARPU

សេវាអនុញ្ញាត CPA

1.10 សេវាអនុញ្ញាត

សេវាអនុញ្ញាត - A/B A/B

1.11 សេវាអនុញ្ញាត

សេវាអនុញ្ញាត A/B A/B

1.12 សេវាអនុញ្ញាត

សេវាអនុញ្ញាត A/B A/B

1.13 亂数生成器

乱数生成器のA/Bテスト結果を表示します。

1.14 データフレーム

データフレームのA/Bテスト結果を表示します。

1.15 モデルパフォーマンス

モデルパフォーマンスのA/Bテスト結果を表示します。

1.16 モデルパフォーマンス

モデルパフォーマンスのA/Bテスト結果を表示します。

1.17 モデルパフォーマンス

モデルパフォーマンスのA/Bテスト結果を表示します。

1.18 モデルパフォーマンス

Precision・Recall・NDCGのモデルパフォーマンスA/Bテスト結果を表示します。

1.19 ဗိုလ်ချုပ်

ဗိုလ်ချုပ်အားလုံး၏ အမြတ်ဆင့် အသေးစိတ် အကြောင်းအရာများ
အမြတ်ဆင့် အသေးစိတ် အကြောင်းအရာများ A/B အနေဖြင့် ပေါ်လောက်ပါသည်

1.20 ဗိုလ်ချုပ်

ဗိုလ်ချုပ်အားလုံး၏ အမြတ်ဆင့် အသေးစိတ် အကြောင်းအရာများ
အမြတ်ဆင့် အသေးစိတ် အကြောင်းအရာများ

1.21 ဗိုလ်ချုပ်

ဗိုလ်ချုပ်အားလုံး၏ အမြတ်ဆင့် အသေးစိတ် အကြောင်းအရာများ
အမြတ်ဆင့် အသေးစိတ် အကြောင်းအရာများ

2 ဗိုလ်ချုပ်

2.1 ဗိုလ်ချုပ်

ဗိုလ်ချုပ်အားလုံး၏ User, Property, Order, Recommendation, Interaction အတွက်
မ Facility, PropertyImage, City, Tag အတွက် အမြတ်ဆင့် အသေးစိတ် အကြောင်းအရာများ
အမြတ်ဆင့် အသေးစိတ် အကြောင်းအရာများ

2.2 ဗိုလ် E-R ဗိုလ်

ဗိုလ်ချုပ်အားလုံး၏ အမြတ်ဆင့် အသေးစိတ် အကြောင်းအရာများ E-R ဗိုလ်

- ဗိုလ်ချုပ်အားလုံး၏ User
- ဗိုလ်ချုပ်အားလုံး၏ Property
- ဗိုလ်ချုပ်အားလုံး၏ Order

- ဗိုလ်ချုပ်
- မြန်မာ

2.3 အောက် E-R ဗြ

အောက်တွင် ပြည်သူ့လုပ်ငန်းမှူးအောင် မြန်မာ-ဗိုလ်ချုပ်-မြန်မာ စီမံချက်များ ဖြစ်ပါသည်။

2.4 အောက် E-R ဗြ

![ER ဗြ](diagrams/er.svg)

E-R ဗြ အတွက် အောက်တွင် ပြည်သူ့လုပ်ငန်းမှူးအောင် မြန်မာ-ဗိုလ်ချုပ်-မြန်မာ စီမံချက်များ ဖြစ်ပါသည်။

2.5 အောက်တွင်

အောက်တွင် ပြည်သူ့လုပ်ငန်းမှူးအောင် မြန်မာ-ဗိုလ်ချုပ်-မြန်မာ စီမံချက်များ ဖြစ်ပါသည်။

အောက်တွင် “မြန်မာ-ဗိုလ်ချုပ်” အောက်တွင် ပြည်သူ့လုပ်ငန်းမှူးအောင် မြန်မာ-ဗိုလ်ချုပ်-မြန်မာ စီမံချက်များ ဖြစ်ပါသည်။ AB ရှိ

3 အောက်တွင်

3.1 E-R အောက်တွင်

အောက်တွင် ပြည်သူ့လုပ်ငန်းမှူးအောင် User မှ Property မှ Order မှ Recommendation မှ Interaction မှ Facility မှ PropertyFacility မှ PropertyImage မှ City မှ Tag မှ မြန်မာ-ဗိုလ်ချုပ်-မြန်မာ စီမံချက်များ ဖြစ်ပါသည်။ PropertyFacility မှ PropertyTag မှ မြန်မာ-ဗိုလ်ချုပ်-မြန်မာ စီမံချက်များ ဖြစ်ပါသည်။

3.2 用 戶 資 料

```
CREATE TABLE user (
    id BIGINT PRIMARY KEY,
    username VARCHAR(50),
    password_hash VARCHAR(255),
    phone VARCHAR(20),
    email VARCHAR(100),
    role VARCHAR(20),
    status VARCHAR(20),
    created_at DATETIME,
    updated_at DATETIME
)
```

3.3 物 品 資 料

```
--user--
```

	id		BIGINT		主鍵	
	username		VARCHAR(50)		用 戶名	
	password_hash		VARCHAR(255)		密碼散列值	
	phone		VARCHAR(20)		電話號碼	
	email		VARCHAR(100)		電子郵件地址	
	role		VARCHAR(20)		角色 (USER/LANDLORD/ADMIN)	
	status		VARCHAR(20)		狀態	
	created_at		DATETIME		創建時間	
	updated_at		DATETIME		最後修改時間	

```
--property--
```

	id		BIGINT		主鍵	
	title		VARCHAR(200)		標題	
	city		VARCHAR(50)		城市	
	address		VARCHAR(200)		地址	
	price		DECIMAL(10,2)		價格	
	bedrooms		INT		臥室數	
	max_guests		INT		最多入住人數	

| rating | DECIMAL(3,2) | 评分 |

| landlord_id | BIGINT | 房东 ID |

| status | VARCHAR(20) | 状态 |

| created_at | DATETIME | 创建时间 |

房屋order

| --- | --- | --- |

| --- | --- | --- |

| id | BIGINT | 订单ID |

| user_id | BIGINT | 用户ID |

| property_id | BIGINT | 房屋ID |

| check_in | DATE | 入住日期 |

| check_out | DATE | 离住日期 |

| total_price | DECIMAL(10,2) | 总价 |

| status | VARCHAR(20) | 状态 |

| created_at | DATETIME | 创建时间 |

用户interaction

| --- | --- | --- |

| --- | --- | --- |

| id | BIGINT | 交互ID |

| user_id | BIGINT | 用户ID |

| property_id | BIGINT | 房屋ID |

| action | VARCHAR(20) | 行为(MVIEW/FAVOR/BOOK) |

| weight | INT | 权重 |

| occurred_at | DATETIME | 发生时间 |

推荐recommendation

| --- | --- | --- |

| --- | --- | --- |

id BIGINT ID
user_id BIGINT 用户ID
property_id BIGINT 房产ID
score DECIMAL(6,4) 分数
strategy VARCHAR(50) 策略
created_at DATETIME 创建时间

房产设施表facility

ID 名称 Facility Name
--- --- ---
id BIGINT ID
name VARCHAR(50) 名称
category VARCHAR(50) 分类
description VARCHAR(200) 描述

房产设施关联表property_facility

ID 房产ID Property ID
--- --- ---
id BIGINT ID
property_id BIGINT 房产ID
facility_id BIGINT 设施ID
created_at DATETIME 创建时间

房产图片表property_image

ID 图片URL Image URL
--- --- ---
id BIGINT ID
property_id BIGINT 房产ID
url VARCHAR(255) 图片地址
sort_order INT 排序

| created_at | DATETIME | 时刻 |

时刻
YYYY-MM-DD HH:MM:SS

3.4 时刻

时刻
YYYY-MM-DD HH:MM:SS PENDING/CONFIRMED/CANCELLED/COMPLETED

时刻
YYYY-MM-DD HH:MM:SS DRAFT/REVIEWING/ACTIVE/INACTIVE

时刻

3.5 时刻

时刻
YYYY-MM-DD HH:MM:SS (user_id, created_at) 时刻
YYYY-MM-DD HH:MM:SS (property_id, occurred_at) 时刻
user.username user.phone

4 时刻

4.1 MySQL

MySQL
CREATE DATABASE homestay_recommendation DEFAULT CHARACTER SET utf8mb4;

时刻
utf8mb4

4.2 MySQL

时刻

CREATE TABLE property (

```
id BIGINT PRIMARY KEY AUTO_INCREMENT,  
title VARCHAR(200) NOT NULL,  
city VARCHAR(50) NOT NULL,  
address VARCHAR(200) NOT NULL,  
price DECIMAL(10,2) NOT NULL,  
bedrooms INT NOT NULL,  
max_guests INT NOT NULL,  
landlord_id BIGINT NOT NULL,  
rating DECIMAL(3,2) DEFAULT 0,  
status VARCHAR(20) DEFAULT 'ACTIVE',  
created_at DATETIME DEFAULT CURRENT_TIMESTAMP  
);
```

4.3 □□□□

```
CREATE VIEW view_hot_property AS  
  
SELECT property_id, COUNT(*) AS order_count  
  
FROM `order`  
  
GROUP BY property_id;
```

4.4

city\price\rating

```
CREATE INDEX idx_property_city ON property(city);  
  
CREATE INDEX idx_property_price ON property(price);  
  
CREATE INDEX idx_property_rating ON property(rating);
```

user_id □ property_id □ (property_id, check_in)□

4.5

A horizontal row of 15 empty square boxes, likely used for a survey or form.

```
CREATE TRIGGER trg_update_rating AFTER INSERT ON review
FOR EACH ROW
BEGIN
UPDATE property
SET rating = (SELECT AVG(score) FROM review WHERE property_id = NEW.property_id)
WHERE id = NEW.property_id;
END;
```

A horizontal row of 30 small, empty rectangular boxes arranged in a single row.

4.6

A horizontal row of fifteen empty rectangular boxes, each with a thin black border, intended for handwritten names.

```
CREATE PROCEDURE sp_get_recommendations(IN uid BIGINT)

BEGIN

SELECT property_id, score

FROM recommendation
```

```
WHERE user_id = uid  
ORDER BY score DESC  
LIMIT 20;  
END;
```

A horizontal row of 30 small, empty rectangular boxes arranged in a single row.

4.7

InnoDB
InnoDB

4.8 □□□□□□□□

5

5.1 □□□□

10

```
SELECT * FROM property  
WHERE city = '□□'  
ORDER BY rating DESC  
LIMIT 10;
```

```
SELECT * FROM `order`  
WHERE user_id = 1001  
ORDER BY created_at DESC;
```

3 

```
SELECT * FROM interaction  
  
WHERE property_id = 2001 AND action = 'VIEW'  
  
ORDER BY occurred_at DESC;
```

```
SELECT p.*, r.score
FROM recommendation r
JOIN property p ON r.property_id = p.id
WHERE r.user_id = 1001
ORDER BY r.score DESC
LIMIT 20;
```

5.2 □□□□

- □□□□□

```
INSERT INTO `order` (user_id, property_id, check_in, check_out, total_price, status)  
VALUES (1001, 2001, '2026-03-01', '2026-03-05', 1200.00, 'PENDING');
```

- □□□□□□□

UPDATE property

SET price = 499.00

WHERE id = 2001;

- ဗိုလ်ချုပ်

INSERT INTO interaction (user_id, property_id, action, occurred_at)

VALUES (1001, 2001, 'BOOK', NOW());

- ဗိုလ်ချုပ်

UPDATE `order`

SET status = 'CANCELLED'

WHERE id = 3001;

ဗိုလ်ချုပ်

5.3 ဗိုလ်ချုပ်

ဗိုလ်ချုပ်

START TRANSACTION;

UPDATE property_calendar

SET available = 0

WHERE property_id = 2001 AND date BETWEEN '2026-03-01' AND '2026-03-05';

INSERT INTO `order` (user_id, property_id, check_in, check_out, total_price, status)

VALUES (1001, 2001, '2026-03-01', '2026-03-05', 1200.00, 'PENDING');

COMMIT;

5.4

is_deleted

6

6.1 □□□□

6.2 □□□□□□

- 
 - 
 - 
 -  SQL
 - 
 - 

6.3 □□□□□□

6.4 项目实践

通过本章的项目实践，你将掌握如何使用 Spring Boot、MyBatis-Plus 和 MySQL 构建一个 RESTful API。同时，你也将学习到如何使用 Vue 3 和 Element Plus 来构建前端界面。

7 项目实践

7.1 项目概述

本章的项目实践将构建一个 RESTful API，该 API 将使用 Spring Boot、MyBatis-Plus 和 MySQL 构建。Vue 3 和 Element Plus 将用于构建前端界面。Spring Boot 将处理后端逻辑，MyBatis-Plus 将与 MySQL 数据库交互。

7.2 项目结构

- **模块**：JWT 处理、Axios 用于发送 Token
- **模块**：前端界面
- **模块**：后端逻辑
- **模块**：数据库交互
- **模块**：集成测试

7.3 项目实践

通过本章的项目实践，你将掌握如何使用 Spring Boot、MyBatis-Plus 和 MySQL 构建一个 RESTful API。同时，你也将学习到如何使用 Vue 3 和 Element Plus 来构建前端界面。

通过本章的项目实践，你将掌握如何使用 Spring Boot、MyBatis-Plus 和 MySQL 构建一个 RESTful API。同时，你也将学习到如何使用 Vue 3 和 Element Plus 来构建前端界面。

7.4 API ໝາຍເຫດ

ໝາຍເຫດ API ອີ່ `/api/auth` `api/property` `api/order` `api/recommendation` ໂດຍມີການຕະຫຼາດການສ້າງຂອງລັບອຳນວຍ

7.5 ມືການ

ມືການ Maven ມີການ `mvn spring-boot:run` ມືການ Vite ອີ່ `npm run build` ມືການ MySQL 8.x ມືການ Redis ມືການ Linux + Docker ມືການ JWT ມືການ ELK ແລະ Prometheus + Grafana ມືການ

7.6 ມືການ

ມືການ Element Plus ມືການ

7.7 ມືການ

ມືການ JWT ມືການ ELK

8 ມືການ

8.1 ມືການ

ມືການ

8.2 ມືການ

Postman 亂 Rest Client 亂 API 亂 Token 亂 401 亂 403 亂 400 亂 500 亂

8.3 亂

JMeter 亂 95% 亂 SQL 亂

8.4 亂

乱

8.5 亂

| 亂 | 亂 | 亂 | 亂 |

| --- | --- | --- | --- |

| TC-01 | 亂 | 亂 | 亂 Token 亂 |

| TC-02 | 亂 | 亂=乱<500 | 亂 |

| TC-03 | 亂 | 亂 | 亂 |

| TC-04 | 亂 | 亂 | 亂 |

| TC-05 | 亂 | 亂 | 亂 401 |

乱

9 亂

乱

10

- [1] Resnick P, Varian H R. Recommender systems. Communications of the ACM, 1997.
 - [2] Sarwar B, et al. Item-based Collaborative Filtering Recommendation Algorithms. WWW, 2001.
 - [3] He X, et al. Neural Collaborative Filtering. WWW, 2017.
 - [4] 陈雷. 推荐系统. 电子工业出版社, 2016.
 - [5] 周志华. 机器学习. 清华大学出版社, 2019.

11.1

200+ 1000+ 300+ 1000+

11.2

11.3 □□□□□□□

执行命令 `npm run build` 构建项目，执行命令 `mvn spring-boot:run` 启动应用，执行命令 `sql/schema.sql` 执行 `schema.sql` 脚本，执行命令 `sql/sample_data.sql` 执行 `sample_data.sql` 脚本。

11.4 亂数生成器

乱数生成器の評価指標 Precision@K, Recall@K, NDCG の実装と A/B テストの実行手順

11.5 モデル評価

モデル評価の実装と A/B テストの実行手順

11.6 データ収集

データ収集の実装と A/B テストの実行手順

11.7 データ分析

データ分析の実装と A/B テストの実行手順

11.8 A/B テスト

A/B テストの実装と A/B テストの実行手順