

# Tank Battle 游戏数据库设计报告

第九组：孙天宇、张泽、袁卓宸

2020 年 5 月 30 日

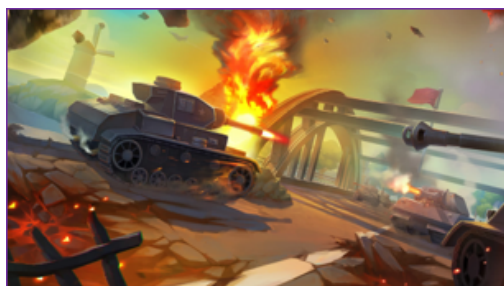
## 目录

<b>1</b>	<b>数据库设计背景</b>	<b>2</b>
1.1	设计主题 . . . . .	2
1.2	数据流图 . . . . .	2
<b>2</b>	<b>ER 图设计说明</b>	<b>3</b>
2.1	实体关系模型设计 . . . . .	3
2.2	各表属性与约束设计 . . . . .	3
<b>3</b>	<b>数据库操作说明</b>	<b>5</b>
3.1	基本的增删改查 . . . . .	5
3.2	复杂查询语句与视图 . . . . .	6
<b>4</b>	<b>可视化界面说明</b>	<b>8</b>
4.1	代码简单说明 . . . . .	8
4.2	操作步骤和功能 . . . . .	9
<b>5</b>	<b>提交文件说明</b>	<b>10</b>

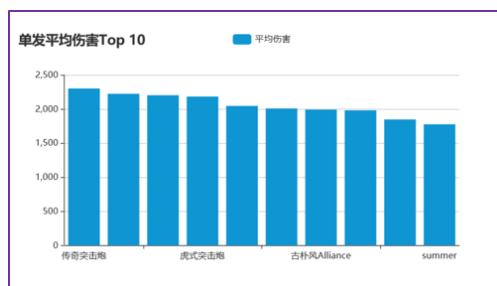
# 1 数据库设计背景

## 1.1 设计主题

我们以网页游戏 Tank Battle(游戏官网<https://tkdz.qq.com/>) 为主题, 设了游戏数据统计分析数据库。该数据库收集了全服范围部分游戏信息, 包含了战斗中的坦克伤害、属性、与战斗结果等数据, 旨在为玩家提供透明真实的游戏数据指导。



(a) 游戏原画鉴赏



(b) 游戏数据统计分析

图 1: 游戏数据库设计背景

## 1.2 数据流图

我们对游戏的 DLL 文件进行反编译 (dnspy), 找到了其中的坦克属性、伤害、战斗结果等消息处理函数, 并让其调用外部我们编写的基于 socket 通信的 DLL, 在游戏过程中将数据实时发送到 java 进程所监听的端口。当 JAVA 进程接受到消息后, 首先对消息类型做判断, 接下来根据其类型进行实时 GUI 显示并通过 JDBC 存入 MySQL 数据库。JAVA 进程还可通过查询数据库中所有的数据信息来对数据做统计分析展示。对于 socket 通信, 我们采用了消息队列与生产者消费者多线程模型来保证数据的正确写入与读取。

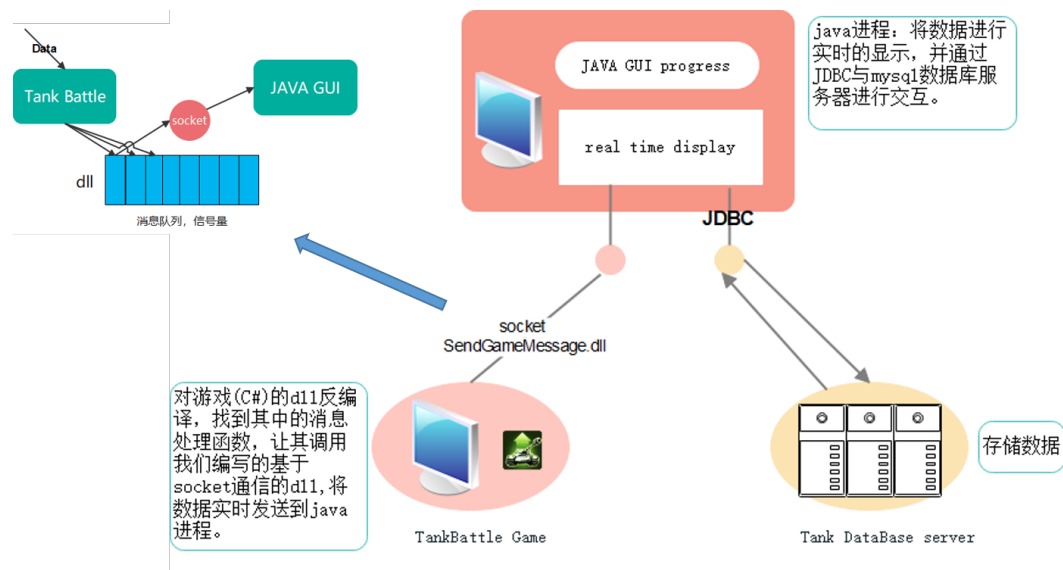


图 2: 整体框架设计

## 2 ER 图设计说明

### 2.1 实体关系模型设计

针对实际获取的游戏数据，我们将实体关系模型设计如图3所示。我们总共设计了六个实体，分别是此软件用户实体、游戏玩家实体、坦克实体、战斗数据实体、战斗结果实体、评论实体。其中软件用户实体与游戏玩家实体是多对一的绑定关系，即多个软件用户可以绑定同一个游戏账号，便于后续的个人游戏战绩统计与分析。每次攻击伤害仅来自于一名玩家，而每个玩家可以有若干条的伤害记录，因此玩家实体与伤害实体为一对多的关系。每次伤害都有一辆攻击坦克与被攻击坦克，因此伤害实体与坦克实体间有两个多对一的攻击与受害的关系。游戏战斗结果与玩家实体、坦克实体是多对一的关系，即一个玩家可以使用一种坦克产生多条的战斗结果数据。此外软件用户实体可以对某一条战斗结果或战斗伤害进行评论。

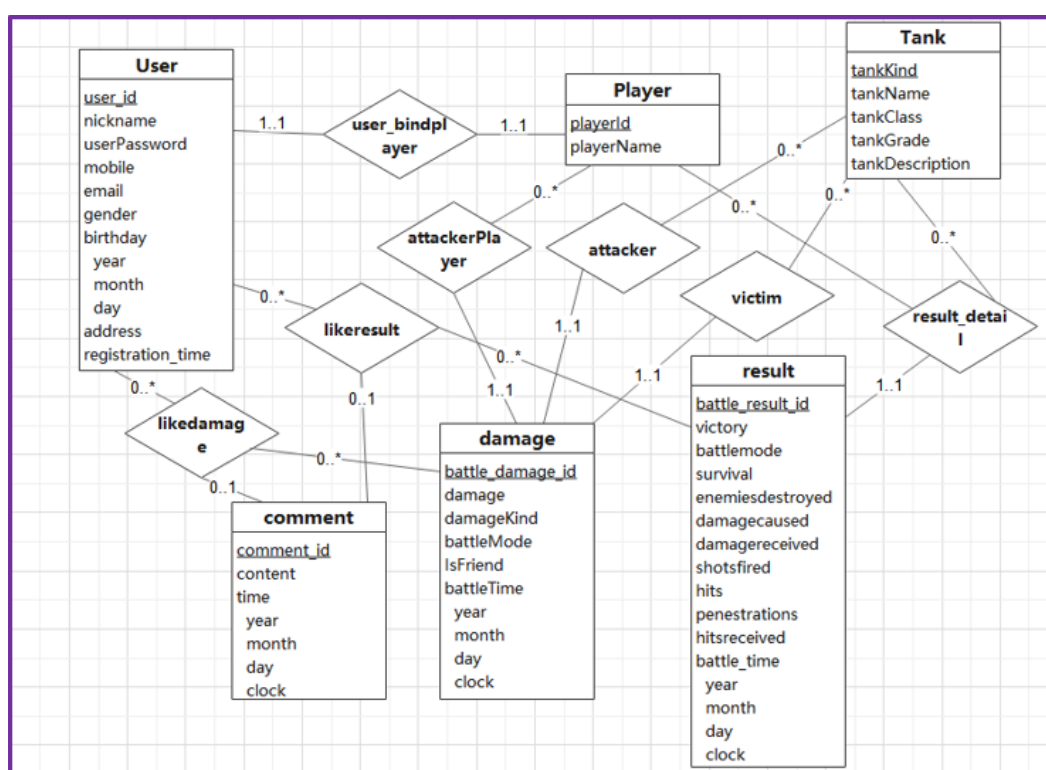


图 3: ER 图设计

### 2.2 各表属性与约束设计

1) 软件用户 user 表，软件用户表记录了此软件的使用者的基本信息，其属性如表1所示，user\_id 为主键。

表 1: 软件使用用户 user 表

user_id	nickname	password	mobile	email	gender	birthday	address	registartion_time
用户 ID	用户昵称	密码	手机	邮箱	性别	生日	地址	注册时间

2) 用户绑定玩家表 (user\_bindplayer), 该表记录了此软件用户绑定游戏账户昵称信息。其中主键为 (user\_id, playerName), 外键约束为 user\_id 参考 user 表, playerName 参考 playerinfo 表。因为软件用户与游戏玩家是一个多对一的关系, 因此设计用户绑定玩家表便于数据库信息维护。

表 2: user\_bindplayer 表

user_id	playerName	bind_time
软件用户 ID	绑定游戏昵称	绑定时间

3) 玩家表 (playerInfo) 该表记录了玩家的游戏的昵称

表 3: user\_bindplayer 表

playerId	playerName
玩家 ID	玩家游戏昵称

4) Tank 表 (tank), 图4所示, 该表记录了从游戏中获取的坦克信息, 其中 tankkind 为主键 (该值为游戏定义), tankName 为坦克名称, tankClass 为坦克类型, tankGrade 为坦克等级, tankDescription 为对坦克的文字描述。

tankkind	tankName	tankClass	tankGrade	tankDescription
312	没钱花-XY		2	8 拥有优秀的侧面防护, 重坦中顶级的速度、射速。Happy【没钱花】【华北一区】的专属座驾
313	蜜糖糖果99式		2	8 集攻击、防御、机动、侦查为一体的优秀重坦! 蜜糖糖果【华南三区】的专属座驾。
316	古朴风格79式		2	8 集攻击、防御、机动、侦查为一体的重坦! 巅峰之古朴风格【华东二区】的专属座驾。
317	十虎狼+89		3	8 需要走位和射击技巧的支援型重火力反坦克炮! 十虎狼+【华南二区】的专属座驾。
318	古朴风格89		3	8 全面超越PTZ-89! 巅峰之古朴风格【华东二区】的专属座驾。
319	泰坦		2	8 正面的防护和火力都是非常恐怖的存在! 但机动较弱!
320	荣耀		2	8 牺牲防御提升了机动, 双连发爆发型重坦!
323	王者121B		1	8 天梯王者专属坦克!
324	皇月神兔		1	8 中坦, 要与正义专属订制的粉色系! 外表清新可人, 内在狂暴小野兽。丕魔族\☆STAR☆【华北一区】的专属座驾
325	醉饮狂歌白羊座		2	8 黄道十二宫Aries白羊座, 《精英》联盟, 醉饮狂歌【华东一区】的专属座驾

图 4: tank 表与部分数据

5) 战斗伤害表 (battleDamage), 6该表记录了玩家使用某坦克在战场上造成的单发伤害信息。其中 battle\_damage\_id 为单次攻击伤害 id, attackerPlayerName 为攻击玩家昵称, attackerKind 为攻击坦克类型, victimeKind 为受害坦克类型, damage 为伤害值, damageKind 为伤害类型, battleMode 为此次伤害所处的游戏模式, IsFriend 表示是否为友军伤害, 属性 battleTime 为造成伤害的时间。该表的主键为 battle\_damage\_id, 外键 attackerPlayerName 参考 PlayerInfo 表, 外键 attackerKind 与 victimKind 参考 tank 表 tankkind 字段。

battle_damage_id	attackerPlayerName	attackerKind	victimKind	damage	damageKind	battleMode	battleId	IsFriend	battleTime
1	15892108901	100	174	680	1	0	66461	1	2020-05-12 21:13:18
2	【龙旗】一能哥	174	110	402	1	0	66461	1	2020-05-12 21:13:23
3	☆南三☆英雄☆舜帝	333	324	1042	1	0	66461	0	2020-05-12 21:13:24
4	づい露薄雲天づい淡雅风致	341	126	388	1	0	66461	0	2020-05-12 21:13:24
5	【龙旗】一能哥	174	341	174	1	0	66461	1	2020-05-12 21:13:27
6	∞兄弟盟∞☆死神☆	174	174	673	9	0	66461	1	2020-05-12 21:13:28
7	15892108901	100	174	639	17	0	66461	0	2020-05-12 21:13:29
8	【龙旗】一能哥	174	341	447	1	0	66461	1	2020-05-12 21:13:30

图 5: 战斗伤害表与部分数据

6) 战斗结果表 (battleResult), 该表记录了某个玩家使用坦克的战斗结果的详细数据信息。其中 battleResult\_id 为战斗结果 id(主键), victory 表示战斗是否胜利, survival 为玩家坦

克是否存活，enemiesdestroyed 为该玩家本场摧毁的坦克数量，damagecaused 为玩家本场战斗总伤害值，damagereceived 为本场战斗的承受伤害，shotsfired 为开火次数，hits 为命中次数，penetrations 为穿透次数，battlemode 表示游戏模式，属性 battletime 为结算时间。该表的主键为 BattleResult\_id 唯一标识某位玩家某一时刻的战斗结果。外键 PlayerName 参考玩家表，tankKind 参考 tank 表，使用外键可以消除玩家与坦克其他信息的冗余，且满足第三范式。

battleResult_id	playerName	tankkind	victory	survival	enemiesdestroyed	damagecaused	damagereceived	shotsfired	hits	penetrations	hitsreceived	battlemode	battletime
5094	Z谢苗·米克尔森	323	1	0	1	3210	3126	8	7	7	6	0	2020-05-15 23:22:18
5095	天马行空666	129	1	0	0	0	3088	1	0	0	4	0	2020-05-15 23:22:18
5097	天下牧业	218	1	0	1	2057	4330	15	8	7	5	0	2020-05-15 23:22:18
5098	SSSSSSSS3170	164	0	1	3	6938	2548	22	14	14	6	0	2020-05-15 23:22:18
5099	小无畏	69	0	1	1	2963	0	8	3	3	0	0	2020-05-15 23:22:18
5100	青春永驻0306	90	0	1	1	3863	0	16	5	5	0	0	2020-05-15 23:22:18
5101	l·卡尔文·爱德华兹	324	0	0	0	897	3088	3	1	1	4	0	2020-05-15 23:22:18
5102	舞翩跹	310	0	0	1	6654	3424	7	7	7	7	0	2020-05-15 23:22:18
5103	夜魔托马斯	145	0	0	1	2190	3893	3	2	2	5	0	2020-05-15 23:22:18
5104	为你倾尽天下，	145	0	0	0	1833	3893	6	2	2	10	0	2020-05-15 23:22:18
5105	一亩五厘	147	1	0	3	17833	5765	9	9	7	7	0	2020-05-15 23:25:04

图 6: 战斗结果表与部分数据

7) 战斗评论表 (userCommentResult)，该表记录了软件用户对某条战斗结果的评价。外键 user\_id 参考 user 表，BattleResult\_id 参考 BattleResult 表。

表 4: userCommentResult 表

user_id	battleResult_id	comment	comment_time
软件用户 ID	战斗结果 ID	评论内容	评论时间

### 3 数据库操作说明

#### 3.1 基本的增删改查

为了便于小组间协作，我们的 MySQL 数据库位于阿里云服务器，IP 地址 123.56.118.161，端口为 3306。用户名为 root，密码为 123456。限于篇幅的原因，基本增删改查分别举一例进行介绍。

插入语句: 此软件需要用户注册后才能够使用，向数据库中添加用户信息。

```
INSERT INTO `user`(`nickname`,`userPassword`,`mobile`,`email`,`gender`,`
    birthday`,`address`) VALUES
('user2', '123456', '13121265866', 'abletian@gmail.com', '男', NULL, NULL),
('user3', '123456', '15548972058', '123084784@qq.com', '女', NULL, NULL)
```

删除语句: 玩家对自己的战斗结果数据进行美化，删除自己零伤的战斗记录。

```
DELETE FROM `battleResult`
WHERE `playerName` = '我要空降皇后（无情）' AND `damagecaused` = 0;
```

更改语句: 软件用户对用户信息进行更新

```
UPDATE user SET `nickname` = '这个杀手有点冷', mobile = '13121265866'
WHERE `user_id` = 1
```

查询语句: 查询某软件用户绑定的游戏玩家昵称信息。

```
SELECT playerName FROM `user`
INNER JOIN user_bindplayer ON `user`.user_id = user_bindplayer.user_id
WHERE nickname = '空降皇后无情'
```

## 3.2 复杂查询语句与视图

游戏数据实时写入数据库均通过底层的 API 调用自动完成，截至目前，数据库已经导入近六万条数据。数据库的复杂操作主要针对坦克数据查询以及用户数据查询，现将主要功能列举如下：

1) 使用 battleDamage 数据查询 PVP 模式下各种坦克的暴击率，并按照降序排列取前 20 名，其中 damageKind 为 9 表示暴击，1 为普通攻击。

```
SELECT `tankName`, COUNT(`damageKind` = 9 OR NULL)/ COUNT(`damageKind` IN
(1, 9) OR NULL) as 'criticalrate'
FROM `battleDamage` INNER JOIN `tank` on `battleDamage`.attackerKind = `
tank`.tankkind
WHERE `battleDamage`.battlemode = 0
GROUP BY `battleDamage`.attackerKind
ORDER BY `criticalrate` desc
LIMIT 20
/*运用battleDamage数据查询PVP模式下各种坦克的暴击率*/
```

2) 使用 battleDamage 数据查询 PVP 模式下各种坦克的单发平均伤害，并进行排名显示。

```
SET @rownum=0;
SELECT @rownum:=@rownum+1 as rownum, `TB`.*
FROM(
SELECT `tankName`, AVG(damage) as 'Average damage per shot'
FROM `battleDamage` INNER JOIN `tank` on `battleDamage`.attackerKind = `
tank`.tankkind
WHERE `battleDamage`.battlemode = 0
GROUP BY `battleDamage`.attackerKind
ORDER BY `Average damage per shot` desc
LIMIT 20
) As `TB`
/*运用battleDamage数据查询PVP模式下各种坦克的单发平均伤害/
```

3) 与上面两个查询类似，我们还实现了命中率、击穿率、单发平均伤害、单发最高伤害、坦克胜率、存活率、单场伤害、场均伤害、单场受到伤害、场均受到伤害、场均被击中次数、场均击毁敌人数等全服坦克信息的查询，详情请参考提交的查询 sql 文件。

4) 查询某一用户所绑定玩家的全部游戏记录。在该查询中我们创建了用户比赛信息视图 userbattleinfo，便于查询。

```
CREATE VIEW userbattleinfo AS
SELECT
`battleResult`.`playerName` AS `playerName`,
`user_bindplayer`.`user_id` AS `user_id`,
`tank`.`tankName` AS `tankName`,
`battleResult`.`tankkind` AS `tankKind`,
```

```

    `battleResult`.`victory` AS `victory`,
    `battleResult`.`survival` AS `survival`,
    `battleResult`.`enemiesdestroyed` AS `enemiesdestroyed`,
    `battleResult`.`damagecaused` AS `damagecaused`,
    `battleResult`.`damagereceived` AS `damagereceived`,
    `battleResult`.`shotsfired` AS `shotsfired`,
    `battleResult`.`hits` AS `hits`,
    `battleResult`.`penetrations` AS `penetrations`,
    `battleResult`.`hitsreceived` AS `hitsreceived`,
    `battleResult`.`battlemode` AS `battlemode`,
    `battleResult`.`battletime` AS `battletime`
FROM
    (
        (
            `battleResult`
            JOIN `user_bindplayer` ON (
                (
                    `battleResult`.`playerName` = `user_bindplayer`.`playerName`
                )
            )
        )
        JOIN `tank` ON (
            (
                `battleResult`.`tankkind` = `tank`.`tankkind`
            )
        )
    )
ORDER BY
    `battleResult`.`battletime` DESC
/*创建用户比赛信息视图userbattleinfo，便于查询/

```

```

SELECT *
FROM `userbattleinfo`
where user_id=7
/*查询某一用户的全部游戏记录/

```

5) 查询用户绑定的玩家使用不同坦克的战斗情况，包含每辆坦克的出战次数、胜率、存活率、场均伤害、场均承受伤害、场均被命中次数、场均击毁数、场均命中率、场均穿透率，并按照出战次数降序排列。图7为某一玩家的查询显示结果。

```

SELECT tankName, COUNT(tankkind) as `usingtimes`,
COUNT(`victory`=0 or Null)/COUNT(tankkind) as 'win rate',
COUNT(`survival`=1 or Null)/COUNT(tankkind) as 'survival rate',
AVG(`damagecaused`) as 'average damage casued',
AVG(`damagereceived`) as 'average damage received',
AVG(`hitsreceived`) as 'average hits received',
AVG(`enemiesdestroyed`) as 'average enemies destroyed',
SUM(`hits`)/SUM(`shotsfired`) as 'accuracy',
SUM(`penetrations`)/SUM(`shotsfired`) as 'penetration rate'
FROM `userbattleinfo`

```



```
WHERE `user_id`=7
GROUP BY `tankkind`
ORDER BY `usingstimes` desc;
/*查询某一用户的坦克情况/
```

tankName	usingstimes	win rate	survival rate	average damage cas	average damage rec	average hits received	average enemies destrc	accuracy	penetration rate
summer	69	0.5507	0.3478	10005.5652	6978.942	6.9855	0.971	0.861	0.7888
传奇59	59	0.6441	0.3729	7537.678	4723.9492	5.2034	1.1695	0.7961	0.7381
巨犀座	39	0.5897	0.4615	8110.9231	6055.1795	6.8205	1.1026	0.8132	0.7356
传奇79P	31	0.5484	0.2258	4591.3548	6644	7.9355	0.6452	0.77	0.625
传奇查狄伦	23	0.5652	0.3913	5998.7391	5600.3913	6.1304	0.6957	0.7214	0.6212
全国联真部落坦克	19	0.2632	0.0526	19705.1579	12043.7895	12.2105	2.7368	0.7913	0.7282
传奇E-100	17	0.4706	0.0588	3551.5294	11088.2941	16.1176	0.2941	0.7632	0.6974
梁山宗师巴顿	14	0.6429	0.1429	4682.9286	5629	6.5714	0.5714	0.6696	0.6522
军团铂金59	7	0.4286	0.1429	2242.7143	6012.2857	5.7143	0.1429	0.6176	0.5294
醉饮狂歌白羊座	7	0.4286	0	2416.5714	7967.2857	9.2857	0.1429	0.6429	0.6071
Azrael	4	0.75	0.75	5008.5	3901.5	5.25	1.5	0.5682	0.5227
老白马天秤座	3	0.6667	0.3333	4560.6667	4503.6667	4.6667	0.6667	0.7391	0.6522
T-59D	3	0.3333	0.3333	3766.6667	4160	7	0.3333	0.8333	0.8333
烈猛T110E4	2	1	0	1300	3947.5	4.5	0	1	1
名人堂T110E5	1	1	0	3679	7926	14	1	0.8	0.8

图 7: 某玩家使用不同坦克的出战情况

## 4 可视化界面说明

图11所示为 JAVA GUI 界面的代码文件树状图。lib 目录为 java 所需要的外部库，如 JDBC 等。resource 文件夹为 GUI 界面所需要使用的图片。src 文件夹为界面的 java 源码。

对于接收到的游戏数据总共设计了三个类，BattleResult.java、Damage.java、TankInfo.java 分别对应坦克基本信息，战斗中的单发伤害信息，战斗结果信息，这三个类将接收到的字符串转化为相应的 java 对象。MysqlAPI.java 完成数据库服务器的连接，并提供将得到的游戏数据写入到数据库的函数接口。DrawDamage.java 对接收到的实时伤害数据做一个动态的显示。ServerProxy.java 对本地端口进行监听，当接收到游戏进程发送过来的数据后，做进一步的处理。TankBox.java 为基本的登录注册界面。TankBoxUserPanel.java 为用户登录后的功能界面，如查看今日战绩、实时伤害数据、全服坦克大数据。

### 4.1 代码简单说明

ServerProxy.java 是一切数据处理的起点，部分核心代码如下，改段代码为当有端口有数据写入后，首先根据数据类型做分类，然后发送到不同的 case 中去执行，有的直接存入数据库，有的则还进行实时显示。

```
while((temp = bw.readLine())!= null)
{
    try{
        if(temp != null){
            String[] result = temp.split(",", 2);
            switch (result[0]){
                case "3":
                    BattleResult br = new BattleResult(temp);
                    if(br.mKind != -1) {
                        MysqlAPI.addBattleResult(br);
                    }
            }
        }
    }
}
```



```

        break;
    case "1":
    case "5":
        TankBoxUserPanel.drawDamage.PushString(temp
        );
        TankBoxUserPanel.drawDamage.repaint();
        TankBoxUserPanel.drawDamage.updateUI();
        Damage damage = new Damage(temp);
        if(damage.mAttackerKind != -1){
            MysqlAPI.addBattleDamage(damage);
        }
        break;
    case "6":
        TankInfo tankinfo = new TankInfo(temp);
        if(tankKinds.contains(Integer.valueOf(
            tankinfo.tankKind))){
            break;
        }else{
            MysqlAPI.addTankInfo(tankinfo);
        }
    default:
        break;
    }
}
}

```

## 4.2 操作步骤和功能

### 1) 注册登录

(a) 注册界面

(b) 登录界面

图 8: 软件注册登录界面

- 2) 实时伤害显示，绿色为友军坦克名称，红色为敌军，后面为伤害与伤害类型。
- 3) 我的今日战绩，基于坦克战斗结果表与当前用户绑定玩家信息，对该玩家本日的战斗数据做统计分析，包含不同坦克场均伤害、出战场次、胜率信息。
- 4) 全服坦克大数据，基于伤害表与战斗结果表，我们统计了全服坦克大数据信息包含暴击率、命中率、击穿率、单发平均伤害、单发最高伤害、坦克胜率、存活率、单场伤害、场均伤害、单场承受伤害、场均承受伤害、场均被击中次数、场均击毁敌人数等。



(a) 实时伤害数据



(b) 我的今日战绩

图 9: 用户登录后界面



图 10: 全服坦克大数据

## 5 提交文件说明

TankStream 文件夹为 JAVA 可视化界面代码

GameStart 文件夹为基于 Windows API 的游戏启动 exe 代码，避免游戏官方启动器的 DLL 文件效应。

DLL\_SendMessage 文件夹为编写的基于 socket 通信的 DLL

大作业报告文件夹包含了报告与报告的 tex 源文件

TankBattle.pptx 为展示 PPT

CreateTable.sql 与 Select.sql 为建表与查询的 SQL 语句

坦克大战源码分析.md 为对游戏源码反编译后修改的记录

整个项目的 github 地址:<https://github.com/sty16/Tank-Battle-Box>

```
lib
├── jna-5.5.0.jar
├── jna-platform-5.5.0.jar
└── mysql-connector-java-8.0.20.jar

out
└── artifacts
    ├── TankStream.jar
    └── TankStream.jar

resources
└── Img
    ├── leftbottom.png
    ├── left.jpg
    ├── rightbottom.png
    └── tank.jpg

src
├── gamedata
│   ├── BattleResult.java
│   ├── Damage.java
│   └── TankInfo.java
├── META-INF
│   └── MANIFEST.MF
├── mysqlapi
│   └── MysqlAPI.java
├── serverproxy
│   └── ServerProxy.java
├── TankBox
│   ├── DrawDamage.java
│   ├── HelloWorld.java
│   ├── TankBox.java
│   ├── TankBox.jfd
│   ├── TankBoxUserPanel.java
│   └── TankBoxUserPanel.jfd
├── tankstream
│   └── TankBoxLogin.java
└── TankStream.iml
```

图 11: java 可视化界面文件说明