

TTkGameDrift API Document (Android)

Name	Date	version	remark
Xuefeng	2020.09.01	V1.1.9	created

1.Version update notes

2. Description

2.1 API call

DriftGame methods are encapsulated in DriftClient. Call it by
DriftClient.getInstance()

2.2 Callback

DriftResultCallback<T>:

All callbacks return the object result of TGResult, which is contained in TGResult :

Response code: int code=result.getResponseCode().

TGResult.isSuccessful() to determine if the call was successful.

The generics T is the return data bean.

Response message: String msg=result.getDebugMessage().

2.3 Field Description

DriftUserInfo :

Params	Type	Note
userId	String	User's id
thirdUid	String	Unique identifier id for third party login
nickname	String	User's nickname
photoUrl	String	User's avatar

score	int	User's current score (playing)
highScore	int	User's highest score
matchesCount	int	Game times with the current user
usedCar	String	User's car
matchesNum	int	User play times
sort	int	User's ranking
state	int	User's status, return value description: 0.send invite 1.accept invite 2. refused invite 3. game over

RoomInfo:

Params	Type	Note
roomId	String	Game room number
homeowner	String	Game host number
playerNum	int	Number of players in the game room
userList	List<DriftUserInfo>	User's ranking
state	int	The state of the game room, return value description: 0.waiting 1.playing 2. over 3. settlement Tips: When the status is >=2, it means that everyone in the room is over

3 API: After Login

3.1 Get User Info

Method	getUserInfo
Params	
callback	DriftResultCallback<DriftUserInfo>
Response	
result	Return status result
userInfo	DriftUserInfo

Example:

```
private void getUserInfo() {
    DriftClient.getInstance().getUserInfo(new
        DriftResultCallback<DriftUserInfo>() {
            @Override
            public void onResult(TGResult result, DriftUserInfo userInfo) {
                if (result.isSuccessful()) {
                    String car= userInfo.usedCar;
                }
            }
        });
}
```

3.2 Update Car Info

Method	updateCarInfo
Params	
car	The id of the car
roomId	The game room id (when in the game room)
callback	DriftResultCallback<DriftUserInfo>
Response	
result	Return status result
userInfo	DriftUserInfo

Example:

```
private void updateCarInfo() {
    DriftClient.getInstance().updateCarInfo("car_001","room_003",new
        DriftResultCallback<DriftUserInfo>() {
            @Override
            public void onResult(TGResult result, DriftUserInfo userInfo) {
                if (result.isSuccessful()) {
                    String uid = userInfo.userId;
                }
            }
        });
}
```

```

    });
}

```

3.3 Get Playing Friends

Method	getPlayingFriends
Params	
callback	DriftResultCallback<List<DriftUserInfo>>
Response	
result	Return status result
friends	List<DriftUserInfo>

Example:

```

private void getUserInfo() {
    DriftClient.getInstance().getPlayingFriends(new
        DriftResultCallback<List<DriftUserInfo>>() {
            @Override
            public void onResult(TGResult result,
                List<DriftUserInfo> friends) {
                if (result.isSuccessful()) {

                }
            }
        })
}

```

3.4 Get Invite List

Method	getInviteList
Params	
callback	DriftResultCallback<List<RoomInfo>>
Response	
result	Return status result
inviteRoomList	List<RoomInfo>

Example:

```

private void getInviteList() {
    DriftClient.getInstance().getInviteList(new
        DriftResultCallback<List<RoomInfo>>() {
            @Override
            public void onResult(TGResult result,
                List<RoomInfo> inviteRoomList) {
                if (result.isSuccessful()) {

                }
            }
        })
}

```

4 API: Play Game

4.1 Create Room

Method	createRoom
Params	
callback	DriftResultCallback<RoomInfo>
userIds	One or more userId who joined the room
Response	
result	Return status result
roomInfo	RoomInfo

Example:

```
private void createRoom() {
    DriftClient.getInstance().createRoom(new DriftResultCallback<RoomInfo>() {
        @Override
        public void onResult(TGResult result, RoomInfo roomInfo) {
            if (result.isSuccessful()) {
                roomId = roomInfo.roomId;
            }
        }
    }, "myUserId", "userId1"...,"userId4");
}
```

4.2 Get Room Info In Waiting

Method	getRoomInfoInWaiting
Params	
roomId	Game room id
callback	DriftResultCallback<RoomInfo>
Response	
result	Return status result
roomInfo	RoomInfo

Example:

```
private void getRoomInfoInWaiting() {
    DriftClient.getInstance().getRoomInfoInWaiting("roomId", new
    DriftResultCallback<RoomInfo>() {
        @Override
        public void onResult(TGResult result, RoomInfo roomInfo) {
            if (result.isSuccessful()) {
                roomState= roomInfo.state;
                // joined the room user list
                List<DriftUserInfo> userList = roomInfo.userList;
            }
        }
    });
}
```

```

    }
    });
}

```

4.3 Start Game

Method	startGame
Params	
roomId	Game room id
callback	DriftResultCallback<RoomInfo>
Response	
result	Return status result
roomInfo	RoomInfo

Example:

```

private void startGame() {
    DriftClient.getInstance().startGame("roomId", new
        DriftResultCallback<RoomInfo>() {
            @Override
            public void onResult(TGResult result, RoomInfo roomInfo) {
                if (result.isSuccessful()) {
                    roomState= roomInfo.state;
                    // joined the room user list
                    List<DriftUserInfo> userList = roomInfo.userList;
                }
            }
        });
}

```

4.4 Update Score

Method	updateScore
Params	
roomId	Game room id
score	user score
callback	DriftResultCallback<RoomInfo>
Response	
result	Return status result
roomInfo	RoomInfo

Example:

```

Handler handler = new Handler();
Runnable updateRunnable = new Runnable() {
    @Override
    public void run() {
        updateScore(roomId, score);
        handler.postDelayed(updateRunnable, 1000 * 3);
    }
}

```

```

    }
};
private void updateScore(String roomId, int score) {
    DriftClient.getInstance().updateScore(roomId, score,
        new DriftResultCallback<RoomInfo>() {
            @Override
            public void onResult(TGResult result, RoomInfo data) {
                if (result.isSuccessful()) {
                    if (data.state == 0) { //waiting game start

                        handler.removeCallbacks(updateRunnable);
                    } else if (data.state == 1) { playing
                        tv_score.setText(score + "");
                    } else if (data.state >= 2) { //game over

                        handler.removeCallbacks(updateRunnable);
                    }
                }
            }
        });
}
}

```

4.5 Game Over

Method	gameOver
Params	
roomId	Game room id
score	user score
callback	DriftResultCallback<RoomInfo>
Response	
result	Return status result
roomInfo	RoomInfo

Example:

```

private void gameOver(String roomId, int score) {
    DriftClient.getInstance().gameOver(roomId, score,
        new DriftResultCallback<RoomInfo>() {
            @Override
            public void onResult(TGResult result, RoomInfo data) {
                if (result.isSuccessful()) { //game over

                }
            }
        });
}
}

```

4.6 Get Room Info In End Game

Method	getRoomInfoInEndGame
Params	
roomId	Game room id
callback	DriftResultCallback<RoomInfo>
Response	
result	Return status result
roomInfo	RoomInfo

Example:

```
private void getRoomInfoInEndGame() {
    DriftClient.getInstance().getRoomInfoInEndGame("roomId", new
        DriftResultCallback<RoomInfo>() {
            @Override
            public void onResult(TGResult result, RoomInfo roomInfo) {
                if (result.isSuccessful()) {
                    roomState= roomInfo.state;
                    // joined the room user list
                    List<DriftUserInfo> userList = roomInfo.userList;
                }
            }
        });
}
```

4.7 Close Room

Method	closeRoom
--------	-----------

5 API: Game Invitation

5.1 Accept Room Invite

Method	acceptRoomInvite
Params	
roomId	Game room id
callback	DriftResultCallback<RoomInfo>
Response	
result	Return status result
roomInfo	RoomInfo

Example:

```
private void acceptRoomInvite() {
    DriftClient.getInstance().acceptRoomInvite("roomId", new
        DriftResultCallback<RoomInfo>() {
            @Override
            public void onResult(TGResult result, RoomInfo roomInfo) {
                if (result.isSuccessful()) {
                    roomState= roomInfo.state;
                    // joined the room user list
                    List<DriftUserInfo> userList = roomInfo.userList;
                }
            }
        });
}
```

5.2 Refuse Room Invite

Method	refuseRoomInvite
Params	
roomId	Game room id
callback	DriftResultCallback<RoomInfo>
Response	
result	Return status result
roomInfo	RoomInfo

Example:

```
private void refuseRoomInvite() {
    DriftClient.getInstance().refuseRoomInvite("roomId", new
        DriftResultCallback<RoomInfo>() {
            @Override
            public void onResult(TGResult result, RoomInfo roomInfo) {
                if (result.isSuccessful()) {
                    roomState= roomInfo.state;
                    // joined the room user list
                    List<DriftUserInfo> userList = roomInfo.userList;
                }
            }
        });
}
```

6 API: Game Rank

6.1 Friends Rank

Method	friendsRank
Params	
callback	DriftResultCallback<List<DriftUserInfo>>
Response	

result	Return status result
userInfoList	List<DriftUserInfo>

Example:

```
private void getFriendsRank() {
    DriftClient.getInstance().friendsRank(
        new DriftResultCallback<List<DriftUserInfo>>() {
            @Override
            public void onResult(TGResult result, List<DriftUserInfo> userInfos) {
                if (result.isSuccessful()) {
                    for (DriftUserInfo userInfo : userInfos) {
                        // ranking of friends
                        int sort = userInfo.sort;
                    }
                }
            }
        });
}
```

6.2 All Rank

Method	allRank
Params	
callback	DriftResultCallback<List<DriftUserInfo>>
Response	
result	Return status result
userInfoList	List<DriftUserInfo>

Example:

```
private void getAllRank() {
    DriftClient.getInstance().allRank(
        new DriftResultCallback<List<DriftUserInfo>>() {
            @Override
            public void onResult(TGResult result, List<DriftUserInfo> userInfos) {
                if (result.isSuccessful()) {
                    for (DriftUserInfo userInfo : userInfos) {
                        // ranking of all user (show up to the top 100)
                        int sort = userInfo.sort;
                    }
                }
            }
        });
}
```