Escrow contract

Brief

- It is a escrow contract where:
    - ☑ players transfer asset temporarily to the contract
    - ☑ contract disburses the card to winner, post the game

About

- contract name - `gpkbatescrow`
- contract's account name - `gpkbatescrow`
- action
    - `transferbypl`
    - `setgstatus`
    - `withdrawbypl`
    - `disburse`
- table
    - `cardwallet`

Compile

```
$ eosio-cpp gpkbatescrow.cpp -o gpkbatescrow.wasm
Warning, empty ricardian clause file
Warning, empty ricardian clause file
Warning, action <transferbypl> does not have a ricardian contract
Warning, action <setgstatus> does not have a ricardian contract
Warning, action <withdrawbypl> does not have a ricardian contract
Warning, action <disburse> does not have a ricardian contract
Warning, action <transferbypl> does not have a ricardian contract
Warning, action <setgstatus> does not have a ricardian contract
Warning, action <withdrawbypl> does not have a ricardian contract
Warning, action <disburse> does not have a ricardian contract
```

Deploy

- deploy contract

```
$ cleosw set contract gpkbatescrow ./
Reading WASM from
/mnt/f/Coding/github_repos/eosio_gpkbattles_contracts/gpkbatescrow/gpkbate
scrow.wasm...
Publishing contract...
executed transaction:
7410d224760874729dc7c3edb04b552c379edeae07d286a8372c6593f160ce87   47592
bytes   6237 us
#          eosio <= eosio::setcode
{"account":"gpkbatescrow","vmtype":0,"vmversion":0,"code":"0061736d0100000
001af022c60000060037f7f7f0...
#          eosio <= eosio::setabi
{"account":"gpkbatescrow","abi":"0e656f73696f3a3a6162692f312e3100050a63617
26477616c6c657400030763617...
warning: transaction executed locally, but may not be confirmed by the
network yet           ]
```

- Adding eosio.code to permissions (for inline actions)

```
$ cleosw set account permission gpkbatescrow active --add-code
executed transaction:
a313929d58e7b07cb4fe2627ad6c25d36d263d462cd264feae12ce485a35e021   184
bytes   173 us
#          eosio <= eosio::updateauth
{"account":"gpkbatescrow","permission":"active","parent":"owner","auth":
{"threshold":1,"keys":[{"key...
warning: transaction executed locally, but may not be confirmed by the
network yet           ]
```

Testing

Action - `transferbypl`

- player `gbuser111121` transfer 3 cards to `gpkbatescrow` contract
  - views the cards owned from `simpleassets` table

```
$ cleosw get table simpleassets gbuser111121 sassets --show-payer
{
  "rows": [{
      "data": {
        "id": "100000000007707",
        "owner": "gbuser111121",
        "author": "gpkbattlesco",
        "category": "exotic",
        "idata": "{}",
        "mdata": "
{\"backimg\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG8e25/back/1.j
pg\",\"cardid\":1,\"img\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG
8e25/base/1a.jpg\",\"name\":\"Joe
Chaotic\",\"quality\":\"a\",\"variant\":\"base\"}",
        "container": [],
        "containerf": []
      },
      "payer": "gpkbatescrow"
    },{
      "data": {
        "id": "100000000007716",
        "owner": "gbuser111121",
        "author": "gpkbattlesco",
        "category": "exotic",
        "idata": "{}",
        "mdata": "
{\"backimg\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG8e25/back/2.j
pg\",\"cardid\":2,\"img\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG
8e25/base/2a.jpg\",\"name\":\"Jettin'
James\",\"quality\":\"a\",\"variant\":\"base\"}",
        "container": [],
        "containerf": []
      },
      "payer": "gpkbatescrow"
    },{
      "data": {
        "id": "100000000007728",
        "owner": "gbuser111121",
        "author": "gpkbattlesco",
        "category": "exotic",
        "idata": "{}",
        "mdata": "
{\"backimg\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG8e25/back/13.
jpg\",\"cardid\":13,\"img\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKi
KG8e25/base/13b.jpg\",\"name\":\"Spiteful
Saffery\",\"quality\":\"b\",\"variant\":\"base\"}",
        "container": [],
        "containerf": []
      },
      "payer": "gpkbatescrow"
    },{
      "data": {
        "id": "100000000007729",
        "owner": "gbuser111121",
```

```
            "author": "gpkbattlesco",
            "category": "exotic",
            "idata": "{}",
            "mdata": "
{\"backimg\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG8e25/back/1.j
pg\",\"cardid\":1,\"img\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG
8e25/base/1b.jpg\",\"name\":\"Schmo
Exotic\",\"quality\":\"b\",\"variant\":\"base\"}",
            "container": [],
            "containerf": []
        },
        "payer": "gpkbatescrow"
      }
    ],
    "more": false,
    "next_key": ""
  }
```

- transfers the owned cards to the contract - `gpkbatescrow` using `transfer`
  action of `simpleassets` contract

```
$ cleosw push action simpleassets transfer '{"from": "gbuser111121",
"to": "gpkbatescrow", "assetids": ["100000000007707",
"100000000007716", "100000000007728"], "memo": "transfer cards for
gpk.battles playing"}' -p gbuser111121@active
executed transaction:
d7f041322eb2b1fe2e4be070c7daee24ff1b223544acd8d9b065174f84329922   216
bytes   704 us
#   simpleassets <= simpleassets::transfer
{"from":"gbuser111121","to":"gpkbatescrow","assetids":
["100000000007707","100000000007716","10000000...
#   gbuser111121 <= simpleassets::transfer
{"from":"gbuser111121","to":"gpkbatescrow","assetids":
["100000000007707","100000000007716","10000000...
#   gpkbatescrow <= simpleassets::transfer
{"from":"gbuser111121","to":"gpkbatescrow","assetids":
["100000000007707","100000000007716","10000000...
#   gpkbattlesc1 <= gpkbattlesc1::empifyplayer
{"asset_contract_ac":"simpleassets","player":"gbuser111121"}
warning: transaction executed locally, but may not be confirmed by the
network yet           ]
```

- view the table `cardwallet` of this contract

```
$ cleosw get table gpkbatescrow gbuser111121 cardwallet --show-payer
{
  "rows": [{
      "data": {
        "card_id": "100000000007707",
        "contract_ac": "simpleassets",
        "usage_status": "available"
      },
      "payer": "gpkbatescrow"
    },{
      "data": {
        "card_id": "100000000007716",
        "contract_ac": "simpleassets",
        "usage_status": "available"
      },
      "payer": "gpkbatescrow"
    },{
      "data": {
        "card_id": "100000000007728",
        "contract_ac": "simpleassets",
        "usage_status": "available"
      },
      "payer": "gpkbatescrow"
    }
  ],
  "more": false,
  "next_key": ""
}
```

- view the table `players` of `gpkbattlesc1` contract

```
$ cleosw get table gpkbattlesc1 gpkbattlesc1 players --show-payer
{
  "rows": [{
      "data": {
        "asset_contract_ac": "simpleassets",
        "players_list": [
          "gbuser111121"
        ]
      },
      "payer": "gpkbattlesc1"
    }
  ],
  "more": false,
  "next_key": ""
}
```

- Similarly do it for player-2 info details using `transfer` action of `simpleassets` contract

```
$ cleosw push action simpleassets transfer '{"from": "gbuser111112",
"asset_contract_ac": "simpleassets", "card_ids": ["100000000007693",
"100000000007694", "100000000007695"], "memo": "transfer cards for
gpk.battles playing"}' -p gbuser111112@active
executed transaction:
d626d8d6b2d881c6212adb24b10ebdafdbeeb5a64916b08015468e9e0acecb85   176
bytes   360 us
#   simpleassets <= simpleassets::transfer
{"from":"gbuser111112","to":"gpkbatescrow","assetids":
["100000000007693","100000000007694","10000000...
#   gbuser111112 <= simpleassets::transfer
{"from":"gbuser111112","to":"gpkbatescrow","assetids":
["100000000007693","100000000007694","10000000...
#   gpkbatescrow <= simpleassets::transfer
{"from":"gbuser111112","to":"gpkbatescrow","assetids":
["100000000007693","100000000007694","10000000...
#   gpkbattlesc1 <= gpkbattlesc1::empifyplayer
{"asset_contract_ac":"simpleassets","player":"gbuser111112"}
warning: transaction executed locally, but may not be confirmed by the
network yet              ]
```

- view the table `cardwallet` of this contract

```
$ cleosw get table gpkbatescrow gbuser111112 cardwallet --show-payer
{
  "rows": [{
      "data": {
        "card_id": "100000000007693",
        "contract_ac": "simpleassets",
        "usage_status": "available"
      },
      "payer": "gpkbatescrow"
    },{
      "data": {
        "card_id": "100000000007694",
        "contract_ac": "simpleassets",
        "usage_status": "available"
      },
      "payer": "gpkbatescrow"
    },{
      "data": {
        "card_id": "100000000007695",
        "contract_ac": "simpleassets",
        "usage_status": "available"
      },
      "payer": "gpkbatescrow"
    }
  ],
  "more": false,
  "next_key": ""
}
```

- view the table `players` of `gpkbattlesc1` contract. Here, `gbuser111111` transferred later than `gbuser111112`

```
$ cleosw get table gpkbattlesc1 gpkbattlesc1 players --show-payer
{
  "rows": [{
      "data": {
        "asset_contract_ac": "simpleassets",
        "players_list": [
          "gbuser111112",
          "gbuser111121"
        ]
      },
      "payer": "gpkbattlesc1"
    }
  ],
  "more": false,
  "next_key": ""
}
```

- player-3 `gbuser111113` transfers wrong card combo & gets error:

```
$ cleosw push action gpkbatescrow transferbypl '{"player": "gbuser111113",
"asset_contract_ac": "simpleassets", "card_ids": ["100000000007697",
"100000000007710", "100000000007714"], "memo": "transfer cards for
gpk.battles playing"}' -p gpkbatescrow@active
Error 3050003: eosio_assert_message assertion failure
Error Details:
assertion failure with message: the cards chosen are of different
combination than (2A,1B) OR (1A,2B).
pending console output:
```

Action - `withdrawbypl`

- player `gbuser111111` wants to voluntarily withdraw the card id: `100000000007690`

```
$ cleosw push action gpkbatescrow withdrawbypl '["gbuser111111",
"simpleassets", "100000000007690"]' -p gbuser111111@active
executed transaction:
113d43bc96b87375ba16a8516843b7146c25329728b7fc89c25eac05c233e8f2   160
bytes   416 us
#  gpkbatescrow <= gpkbatescrow::withdrawbypl
{"player":"gbuser111111","asset_contract_ac":"simpleassets","card_id":"100
000000007690"}
#  simpleassets <= simpleassets::transfer
{"from":"gpkbatescrow","to":"gbuser111111","assetids":
["100000000007690"],"memo":"gbuser111111 withd...
#  gpkbatescrow <= simpleassets::transfer
{"from":"gpkbatescrow","to":"gbuser111111","assetids":
["100000000007690"],"memo":"gbuser111111 withd...
#  gbuser111111 <= simpleassets::transfer
{"from":"gpkbatescrow","to":"gbuser111111","assetids":
["100000000007690"],"memo":"gbuser111111 withd...
warning: transaction executed locally, but may not be confirmed by the
network yet             ]
```

- view the table `sassets` of `simpleassets` contract for `gbuser111111` player

```
$ cleosw get table simpleassets gbuser111111 sassets --show-payer
{
  "rows": [{
      "data": {
        "id": "100000000007690",
        "owner": "gbuser111111",
        "author": "gpkbattlesco",
        "category": "exotic",
        "idata": "{}",
        "mdata": "
{\"backimg\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG8e25/back/13.
jpg\",\"cardid\":13,\"img\":\"QmYkMDkB1d8ToHNHnFwpeESF3NpfbPKi
KG8e25/base/13b.jpg\",\"name\":\"Spiteful
Saffery\",\"quality\":\"b\",\"variant\":\"base\"}",
        "container": [],
        "containerf": []
      },
      "payer": "gpkbatescrow"
    }
  ],
  "more": false,
  "next_key": ""
}
```

- view the table `cardwallet` of this contract
```

```
$ cleosw get table gpkbatescrow gbuser111111 cardwallet --show-payer
{
  "rows": [{
      "data": {
        "card_id": "100000000007691",
        "contract_ac": "simpleassets",
        "usage_status": "available"
      },
      "payer": "gpkbatescrow"
    },{
      "data": {
        "card_id": "100000000007692",
        "contract_ac": "simpleassets",
        "usage_status": "available"
      },
      "payer": "gpkbatescrow"
    }
  ],
  "more": false,
  "next_key": ""
}
```

- view the table `players` of `gpkbattlesco` contract

```
$ cleosw get table gpkbattlesc1 gpkbattlesc1 players --show-payer
{
  "rows": [{
      "data": {
        "asset_contract_ac": "simpleassets",
        "players_list": [
          "gbuser111111"
        ]
      },
      "payer": "gpkbattlesc1"
    }
  ],
  "more": false,
  "next_key": ""
}
```

- player `gbuser111111` wants to voluntarily withdraw the card id: `100000000007691`

```
$ cleosw push action gpkbatescrow withdrawbypl '["gbuser111111",
"simpleassets", "100000000007691"]' -p gbuser111111@active
executed transaction:
71727bc60e1a35c9fd0ab894059652e20e1f715300b010a9a9c6ba4443832678   160
bytes   318 us
#  gpkbatescrow <= gpkbatescrow::withdrawbypl
{"player":"gbuser111111","asset_contract_ac":"simpleassets","card_id":"100
000000007691"}
#  simpleassets <= simpleassets::transfer
{"from":"gpkbatescrow","to":"gbuser111111","assetids":
["100000000007691"],"memo":"gbuser111111 withd...
#  gpkbatescrow <= simpleassets::transfer
{"from":"gpkbatescrow","to":"gbuser111111","assetids":
["100000000007691"],"memo":"gbuser111111 withd...
#  gbuser111111 <= simpleassets::transfer
{"from":"gpkbatescrow","to":"gbuser111111","assetids":
["100000000007691"],"memo":"gbuser111111 withd...
warning: transaction executed locally, but may not be confirmed by the
network yet            ]
```

- view the table `sassets` of `simpleassets` contract for `gbuser111111` player

```
$ cleosw get table simpleassets gbuser111111 sassets --show-payer
{
  "rows": [{
      "data": {
        "id": "100000000007690",
        "owner": "gbuser111111",
        "author": "gpkbattlesco",
        "category": "exotic",
        "idata": "{}",
        "mdata": "
{\"backimg\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG8e25/back/13.
jpg\",\"cardid\":13,\"img\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKi
KG8e25/base/13b.jpg\",\"name\":\"Spiteful
Saffery\",\"quality\":\"b\",\"variant\":\"base\"}",
        "container": [],
        "containerf": []
      },
      "payer": "gpkbatescrow"
    },{
      "data": {
        "id": "100000000007691",
        "owner": "gbuser111111",
        "author": "gpkbattlesco",
        "category": "exotic",
        "idata": "{}",
        "mdata": "
{\"backimg\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG8e25/back/3.j
pg\",\"cardid\":3,\"img\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG
8e25/base/3a.jpg\",\"name\":\"Jeff
Joke\",\"quality\":\"a\",\"variant\":\"base\"}",
        "container": [],
        "containerf": []
      },
      "payer": "gpkbatescrow"
    }
  ],
  "more": false,
  "next_key": ""
}
```

- view the table `cardwallet` of this contract

```
$ cleosw get table gpkbatescrow gbuser111111 cardwallet --show-payer
{
  "rows": [{
      "data": {
        "card_id": "100000000007692",
        "contract_ac": "simpleassets",
        "usage_status": "available"
      },
      "payer": "gpkbatescrow"
    }
  ],
  "more": false,
  "next_key": ""
}
```

- view the table `players` of `gpkbattlesco` contract

```
$ cleosw get table gpkbattlesc1 gpkbattlesc1 players --show-payer
{
  "rows": [{
      "data": {
        "asset_contract_ac": "simpleassets",
        "players_list": [
          "gbuser111111"
        ]
      },
      "payer": "gpkbattlesc1"
    }
  ],
  "more": false,
  "next_key": ""
}
```

- player `gbuser111111` wants to voluntarily withdraw the last card id:
  `100000000007692`

```
$ cleosw push action gpkbatescrow withdrawbypl '["gbuser111111",
"simpleassets", "100000000007692"]' -p gbuser111111@active
executed transaction:
c42c0bb089734d8fbedb2e35a4203710342459d2ef2b6f69373a483186003287   160
bytes   456 us
#  gpkbatescrow <= gpkbatescrow::withdrawbypl
{"player":"gbuser111111","asset_contract_ac":"simpleassets","card_id":"100
000000007692"}
#  simpleassets <= simpleassets::transfer
{"from":"gpkbatescrow","to":"gbuser111111","assetids":
["100000000007692"],"memo":"gbuser111111 withd...
#  gpkbattlesc1 <= gpkbattlesc1::remplayer
{"asset_contract_ac":"simpleassets","player":"gbuser111111"}
#  gpkbatescrow <= simpleassets::transfer
{"from":"gpkbatescrow","to":"gbuser111111","assetids":
["100000000007692"],"memo":"gbuser111111 withd...
#  gbuser111111 <= simpleassets::transfer
{"from":"gpkbatescrow","to":"gbuser111111","assetids":
["100000000007692"],"memo":"gbuser111111 withd...
warning: transaction executed locally, but may not be confirmed by the
network yet            ]
```

- view the table `sassets` of `simpleassets` contract for `gbuser111111` player

```
$ cleosw get table simpleassets gbuser111111 sassets --show-payer
{
  "rows": [{
      "data": {
        "id": "100000000007690",
        "owner": "gbuser111111",
        "author": "gpkbattlesco",
        "category": "exotic",
        "idata": "{}",
        "mdata": "
{\"backimg\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG8e25/back/13.
jpg\",\"cardid\":13,\"img\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKi
KG8e25/base/13b.jpg\",\"name\":\"Spiteful
Saffery\",\"quality\":\"b\",\"variant\":\"base\"}",
        "container": [],
        "containerf": []
      },
      "payer": "gpkbatescrow"
    },{
      "data": {
        "id": "100000000007691",
        "owner": "gbuser111111",
        "author": "gpkbattlesco",
        "category": "exotic",
        "idata": "{}",
        "mdata": "
{\"backimg\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG8e25/back/3.j
pg\",\"cardid\":3,\"img\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG
8e25/base/3a.jpg\",\"name\":\"Jeff
Joke\",\"quality\":\"a\",\"variant\":\"base\"}",
        "container": [],
        "containerf": []
      },
      "payer": "gpkbatescrow"
    },{
      "data": {
        "id": "100000000007692",
        "owner": "gbuser111111",
        "author": "gpkbattlesco",
        "category": "exotic",
        "idata": "{}",
        "mdata": "
{\"backimg\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG8e25/back/1.j
pg\",\"cardid\":1,\"img\":\"QmYkMDkB1d8ToHNHnFwpeESF3Npfid671NrfbPKiKG
8e25/base/1b.jpg\",\"name\":\"Schmo
Exotic\",\"quality\":\"b\",\"variant\":\"base\"}",
        "container": [],
        "containerf": []
      },
      "payer": "gpkbatescrow"
    }
  ],
  "more": false,
```

```
      "next_key": ""
    }
```

- view the table `cardwallet` of this contract

```
$ cleosw get table gpkbatescrow gbuser111111 cardwallet --show-payer
{
  "rows": [],
  "more": false,
  "next_key": ""
}
```

- view the table `players` of `gpkbattlesco` contract

```
$ cleosw get table gpkbattlesc1 gpkbattlesc1 players --show-payer
{
  "rows": [{
      "data": {
        "asset_contract_ac": "simpleassets",
        "players_list": []
      },
      "payer": "gpkbattlesc1"
    }
  ],
  "more": false,
  "next_key": ""
}
```

- Observation:
  - the players list gets empty when all it's cards are removed from the `cardwallet`

TODO

- ☐ find out way to add `transferbypl` ACTION as `on_notify[["*::transfer"]]` for asset transfer & then add player's details in the `cardwallet` table & `players` table.
- before deploy to MAINNET,
  - ☐ change the card owner's name from `gpkbattlesco` to `gpk.topps`
  - ☐ change the game contract account name from `gpkbattlesc1` to `gpkbattlesco`
- ☐ change the game contract account name from `gpkbattlesc1` to `gpkbattlesco` in the check_gfee_balance() func in `gpkbattlesco` contract