



AP JSON Spec

Authentication

Send GET request to https://master.alphapoker.org/connect?key=your_key where "your_key" is your unique api key (remember to keep this private).

Response format:

```
{
  "success": true,
  "end_date": "2020-08-27",
  "session_id": "94c5a5",
  "start_date": "2018-10-10"
}
```

Hand history format

Send POST request to <https://master.alphapoker.org/hh> with body containing the json.

```
{
  # HEADER
  "data_source": "my app name", # data provider (listener name)
  "source_hand_id": "3720784100002", # unique hand id from poker app
  "game_id": "3720784100002", # same as source_hand_id, for compatibility
  "source_table_id": "320557", # unique table id from poker app
  "source_table_name": "PPP_12_60_320557", # unique table name from poker app, could be same as table id
  "hero_seat": 0, # hero seat
  "seats": 2, # number of players in current game (standing up, zero stack must not be counted)
  "session_id": "530a37", # unique AP session_id, returned in auth json response
  "site": "pppoker", # poker app/site (pokermaster, pokerking, firepoker, etc)
  "table_id": 1, # table_id in UI. 1,2,3,4....
  "table_type": "unknown", # NLH 6-way, NLH 2-way, etc
  "table_name": "Emulator-1", # table name shown in UI
  "return_data": true, # true - return action options as json and show in UI; false - show only in UI

  # GAME ACTIONS
  "actions": [
    {
      "name": "StonyRidge", # user name
      "uid": "123123", # user id
      "seat": 0, # user seat, seats are strictly sequential (eg. seats=3; seat=0,1,2)
      "type": "stack", # stacksize
      "value": 384 # value
    },
    {
      "name": "Vad3r",
      "uid": "56745674", # user id
      "seat": 1,
      "type": "stack",
      "value": 416
    },
    {
      "seat": 0,
      "type": "ante", # ante, if ante game
      "value": 1.0
    },
    {
      "seat": 1,
      "type": "ante",

```

```

    "value": 1.0
  },
  {
    "seat": 1,
    "type": "sb", # small blind
    "value": 2.0
  },
  {
    "seat": 0,
    "type": "bb", # big blind
    "value": 4.0
  },
  {
    "seat": 0,
    "type": "str", # straddle
    "value": 8.0
  },
  {
    "cards": "7s6d", # preflop starts here, dealt hero cards
    "type": "hero_cards"
  },
  {
    "seat": 1,
    "type": "call",
    "value": 2.0
  },
  {
    "seat": 0,
    "type": "check"
  },
  {
    "cards": "4h5dTd",
    "type": "flop"
  },
  {
    "seat": 0,
    "type": "raise",
    "value": 10.0 # raise amount is relative to what player has already in pot
  },
  {
    "seat": 1,
    "type": "call",
    "value": 12.0
  },
  {
    "cards": "8s",
    "type": "turn"
  },
  {
    "seat": 0,
    "type": "check"
  },
  {
    "seat": 1,
    "type": "check"
  },
  {
    "cards": "5s",
    "type": "river"
  },
  {
    "seat": 0,
    "type": "allin"
  },
  {
    "seat": 1,
    "type": "call"
  },
  {"cards": "7s6d", "seat": 0, "type": "show"}, # showdown; skip actions if no-showdown
  {"cards": "Jh8s", "seat": 1, "type": "show"},
  {"value": 123456789, "seat": 0, "pot": 0, "type": "win"}
# win; could be multiple winners, over multiple pots (0, 1, 2, etc).

```

```
]
}
```

Solution format

```
{
  "street": "flop",
  "players": 3, # players in pot
  "pot": 6.0, # pot size
  "size_correction": 0.0, # max player pot size - hero pot size
  "time": 0.5881145119783469, # AP request processing time
  "actions": [
    {
      "freq": 0.22094398736953735, # frequency
      "type": "allin" # action type (allin, raise, call, check, fold)
    },
    {
      "absolute": 1.0285714285714285, # raise size (relative to hero pot)
      "freq": 0.3741452395915985,
      "type": "raise"
    },
    {
      "freq": 0.14701569080352783,
      "type": "check"
    }
  ]
}
```