

# Protokol<sub>i</sub>\_baza

Szymon Witamborski

23 marzec 2010

## Contents

<b>1</b>	<b>Protokół</b>	<b>1</b>
1.1	Inicjowane od strony klienta . . . . .	1
1.2	Inicjowane od strony serwera . . . . .	2
<b>2</b>	<b>Baza/XML</b>	<b>2</b>
2.1	Users . . . . .	2
2.2	Wins . . . . .	3

Opis ideowy, małe litery to dane, WIELKIE to komunikaty.

## 1 Protokół

### 1.1 Inicjowane od strony klienta

(-> do serwera, <- do klienta)

```
-> LOGIN(uname, pass)
    <- WELCOME(sessid)
    <- GET_LOST
-> CREATE_USER(uname, pass)
    tworzy usera i go loguje jesli udalo sie stworzyc
    <- WELCOME(sessid)
    <- GET\_LOST
-> SETAVATAR(blob, typ)
    blob - binarne dane, typ - gif/jpg/png/...
    <- SUCCESS
    <- FAIL(cause) // jakiś meaningful komunikat
```

```

-> SIT(table_id, [pass])
    <- BAD_PASS
    <- IN
-> DECIDE(action, sum)
    action - jedna z akcji możliwych w rozgrywce (check, allin itd.)
    sum - pieniądze skojarzone z tą akcją (ile podbił np.)
    <- FAIL
    <- SUCCESS
-> MESSAGE(content)
    wiadomość na chacie

```

## 1.2 Inicjowane od strony serwera

```

<- TABLE_USERS(login, moneyin, avatar_blob, avatar_typ[, login ...])
    wysyła do klienta informacje o wszystkich graczach
    i z jaką kasą przybyli
<- TABLE_CARD(card)
    pojawienie się na stole nowej karty
<- USERACTION(login, action, sum)
    wysyłane do wszystkich klientów po podjęciu przez aktywnego gracza
    decyzji, treść analogicznie jak w DECIDE
<- WIN(login)
    informacja o wygranym loginie, jednocześnie koniec rozgrywki
    i początek następnej na tym samym stole.
<- SIDEPOT(login, sum[, login, sum, ...])
    informacja o sumach bocznych, dla każdego loginu suma
<- MESSAGE(login, content)
    wiadomość na czacie od użytkownika

```

## 2 Baza/XML

To jest tylko sugestia:

### 2.1 Users

```

int pkey id
vchar login
vchar pass

```

int sum *stan konta*  
int wins *ilość wygranych*  
datetime bankrupt\_date *oznaczane kiedy user zbankrutował, system daje userowi z automatu jakąś kasę raz na dobę jeśli zbankrutował, opisane gdzie indziej dokładnie*  
blob avatar  
vchar avatar\_type

## 2.2 Wins

int pkey id  
int userid  
int sum  
datetime date  
vchar scheme *układ z jakim user wygrał*