

Welcome Page

gameInfo : String
devInfo : String
...

log-In ()
register ()
view Info ()
render ()
...

Register Page

nickName : String
password : String
email : String ...

validate NickName (nickName: String)
validate Email (email: String)
validate Password (password: String)
create Profile (nickName: String, email: String, password: String)
render ()
send To Home Page ()
...

Log-In Page

nickName : String
password : String
...

validate Name And Password (nickName: String, password: String)
accept+Log In ()
render ()
send To Home Page ()
...

HomePage

header : Header
invite : Invitation ...

view Profile ()
view History ()
view Rules ()
view Games ()
continue Game ()
create Invitation (player: Profile)
render ()
...

<<interface>>

Header

logo : image
...

log-Out ()
view Notifications ()
go To Home ()
render ()
sendMessage ()
...

Notification

message : String
sender : Profile
receiver : Profile
invite : Invitation = null
...

<<constructor>> Notification (sender: Profile, receiver: Profile, isInvite: Bool, message: String)
render ()
accept+Invite ()
...

↓
Profile Rules Home Board

HTTP request class
associations

retrieve Match
retrieve Profile

Match

player1: Profile isActive: Bool
player2: Profile isComplete: Bool
~~board: Board~~ numMoves
gameState: JSON winner: string
... endOfGame: Bool = false

State

~~board: Board
...
[[constructor]] State()
saveState()~~

[[constructor]] Match(...)
makeMove() 'String', JSON
recordGame() 2D array
endGame() Match(state)
readJSON()
writeJSON()

Move

~~piece: Piece
...
[[constructor]] Move()
validateMove()
detectEndOfGame()
makeMove()~~

GameResult

~~winner: Profile
loser: Profile
numMoves: int
...
[[constructor]] GameResult(winner: Profile, loser: Profile)~~

Record

~~...
[[constructor]] Record()
addGameResult(game: GameResult)
retrieveGame(...) : GameResult~~

Design ideas Tunes

Header

```
...  
«constructor» Header ()  
log-Out ()  
view Notifications ()  
goTo Home ()  
view Tabs ()
```

Notification

message: String

sender: Profile

receiver: Profile ... invite: Invitation = null

```
«constructor» Notification (sender: Profile, receiver: Profile, is Invite: Bool, message: String)
```

read ()

...

Invitation

sender: Profile

receiver: Profile

accepted By Receiver: Bool = false

accepted By Sender: Bool = True

...

```
«constructor» Invitation (sender: Profile, receiver: Profile)
```

start Game ()

~~accept Invite ()~~

cancel Invite ()

send Invite ()

...

Game Page

board: Board

winner: String = null

player1: Profile

player2: Profile

turn: String

...

game: Match

```
«constructor» Game Page (player1: Profile, player2: Profile)
```

move Piece (piece: Piece)

save Game ()

quit Game ()

Piece

name: String

image: String

...

...

...

Board

ArrayList <piece>

...

image: String

...

