# **Diplomska naloga: Bot za igranje Taroka**

# Tarok

Tarok je družabna igra s kartami, ki ima povsod po Evropi različice. Igra je v Sloveniji močno razširjena po vseh regijah. Cilj igre je zbrati več točk kot ostali soigralci. V Sloveniji je bila leta 2001 ustanovljena Tarok Zveza Slovenije (TARZS).

## Zgodovina

### Zgodovina kart

Prvi zapisi o igralnih kartah za igre obstajajo že od srenjega veka na Kitajskem. “Učenjak Ouyang Xiu (1007 - 1072) je trdil, da so prve igralne karte in igre s kartami obstajale vsaj od sredine dinastije Tang (618 - 907)”. Širjenje kart je tudi močno povezano z izumom tiska in časopisnih aktivnosti. Da je dinastija Tang delila svoje ideje in znanje, je obveščala preko papirja oz. natisnjenih izdelkov. [stran 33, 34]

V Evropi smo dobili prve igralne karte šele v 14. stoletju. Takrat so jih imenovali mameluke. Paket kart je vseboval 52 kart s štirimi barvami: polo palice, kovanci, meči, skodelice. Prvi pisni dokaz o prisotnosti kart v Evropi se nahaja v Bernu v Švici in sicer leta 1367, kjer so jih prepovedali. Prepoved so izdali zaradi iger na srečo.

I

II

XXI

### Zgodovina taroka

Blablebli

## Tarok Zveza Slovenije (TARZS)

TZS

## Pravila Igre

Rules

# Pravila taroka

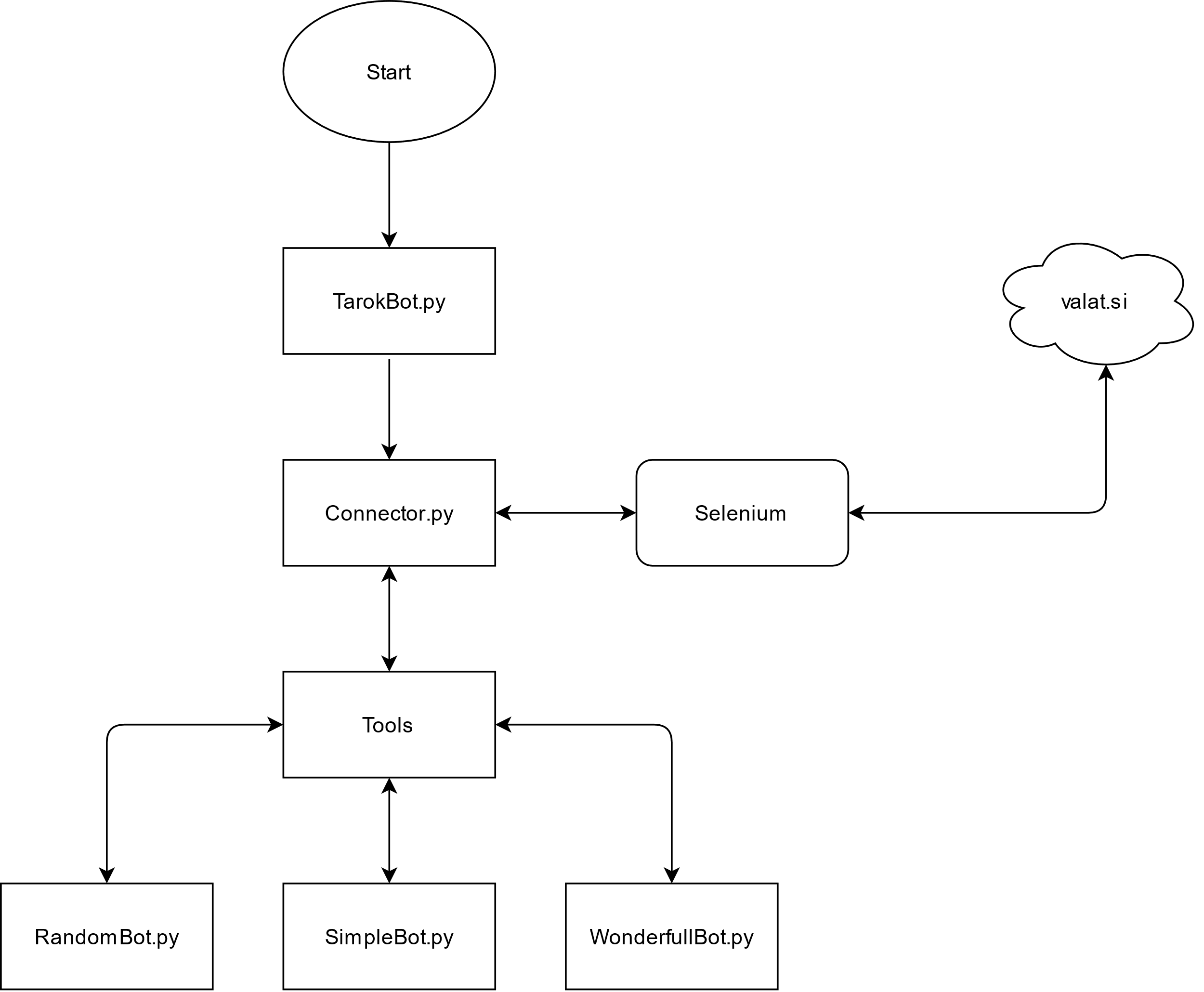
Rules here

# Selenium

Insert selenium doc here

# Portal valat.si

# Logika programa



Za zagon programa je potrebno na začetku v konfiguraciji nastaviti nekatere parametre. Ko je enkrat aplikacija v zagonu se težavnost botov ne da spreminjati. Potrebno je ponovno zagnati aplikacijo. Program je kar sprecifičen za igre v Dve in v Eno.

## Connector

Razred Connector skrbi za komunikacijo med spletno stranjo ‘valat.si’ in boti. S pomočjo Selenium frameworka bere HTML elemente in pripravlja podatke katere bodo boti uporabljali. Glavne metode tega razreda so:

* Choose\_game()
* Choose\_king()
* Choose\_talon()
* Play\_card()

### Connector.choose\_game()

Pri choose\_game metodi bi poudaril, da izbere najvišjo priporočeno igro (‘Ena’ ali ‘Dve’) katera jo predlaga na ‘valat.si’ strani. Zato ni potrebe po logiki pri botih.

### Connector.choose\_king()

Izbere si kralja

### Connector.choose\_talon()

Izbere kupček in si založi karte.

### Connector.play\_card()

Play\_card klika po kartah v roki.

## Tools

Tools razred je komunikator med zagnanim botom in Connector-jem. Pridobi podatke in pretvori v ustrezne razrede za bota. Glavne metode:

* Choose\_king()
* Choose\_talon\_step\_1()
* Choose\_talon\_step\_2()
* Play\_card()

Izbiro talone je bilo potrebno ločiti na 2 koraka zaradi lažje komunikacije med boti in Connector razredom.

# **Boti**

Vsi boti uporabljajo metode opisane pri Tools razredu vendar je logika za vsako metodo pri vsakem botu drugačna.

## RandomBot

Bot, ki večino odločitev naredi naključno.

### Choose\_king()

Naključno izbere med kralji katere nimam v svojih kartah.

### Choose\_talon\_step\_1()

V primeru, da se porufa izbere tisti kupček v katerem je kralj v nasprotnem primeru pa naključno izbere en kupček.

### Choose\_talon\_step\_2()

Naključno si izbere karte za založiti. Karte ne smejo biti kralji ali taroki

### Play\_card()

Naključno izbere karto katero bo odigral.

## SimpleBot

Simple bot vsebuje preprosto začetniško logiko. Ne upošteva oz. ne igra na soigralca ampak igra kot, da bi sam poskušal pobrati čimveč štihov.

### SimpleBot.choose\_king()

Izbere si barvo katere nima v roki. Če ima vse barve v roki potem izbere naključno med tistimi, katerih nimam kraljev v roki.

### SimpleBot.choose\_talon\_step\_1()

V primeru, da se porufa izbere tisti kupček v katerem je kralj v nasprotnem primeru pa izbere kupček z največ točkami.

### SimpleBot.choose\_talon\_step\_2()

Poskuša si založiti barve, da je vsaj 1 barve škrt.

### SimpleBot.play\_card()

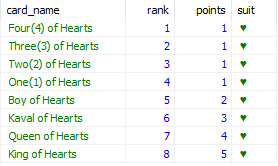
Metoda vrne karto, ki jo določi logika bota. Logika za igranje se razdeli na 3 metode in preverja po temu vrstnemu redu:

* Play\_first(); ko na mizi ni nobene karte in je prvi na vrsti
* Play\_color(); ko je na mizi že neka barva in moram to barvo odigrati.
* Play\_tarot(); ko nimam barve in odigram taroka

### SimpleBot.play\_first()

Naključno izberem karto z barvo. Taroke iz tu izključujem

### SimpleBot.play\_color()



Poiščem najvišjo barvo in najvišjega taroka

Če na mizi ni taroka IN če imam višjo barvo potem odigram naslednjo višjo barvo. V nasprotnem primeru odigram najnižjo barvo.

Primer 1:

V roki imam Three(3) of Hearts, Queen of Hearts in King of Hearts. Igralec nasproti mene je prvi na vrsti in je odigral One(1) of Hearts. Naslednji igralec odigra Kaval of Hearts. Bot bo odigral naslednjo višjo barvo kar je Queen of Hearts.

Primer 2:

V roki imam Three(3) of Hearts, Queen of Hearts in King of Hearts. Igralec nasproti mene je prvi na vrsti in je odigral One(1) of Hearts. Naslednji igralec odigra tarok XII. Bot bo odigral najnižjo barvo kar je Three(3) of Hearts.

### SimpleBot.play\_tarot()

Poiščem najvišji tarok. Če lahko grem čez njega potem odigra naslednji najvišji tarok. V nasprotnem primeru odigra najnižji tarok.

Primer 1:

V roki imam V, XVI in XXI. Igralec nasproti mene je prvi na vrsti in je odigral VII. Naslednji igralec odigra XVII. Bot bo odigral naslednji višji tarok kar je XXI ne glede na to ali je škis že padel ali ne.

Primer 2:

V roki imam V, XVI in XXI. Igralec nasproti mene je prvi na vrsti in je odigral VII. Naslednji igralec odigra XVII. Bot bo odigral naslednji višji tarok kar je XXI ne glede na to ali je škis že padel ali ne.

V primeru da v svojih kartah nima več tarokov vendar bi ga moral dati potem vsakič odigra najmanj vredno karto.

## WonderfulBot

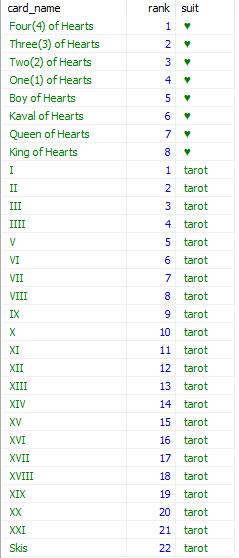
WonderfulBot vsebuje kompleksnejšo logiko. Ta upošteva oz. igra na soigralca in med igro skuša dati čimveč točk soigralcu.

### WonderfulBot.choose\_king()

Psevdokoda logike

<https://snipsave.com/user/simonoblak/snippet/Inbu37pElUxObVip2H/>

### WonderfulBot.choose\_talon\_step\_1()



Psevdokoda logike

<https://snipsave.com/user/simonoblak/snippet/e4Je8pDYAALbnT0Hlj/>

### WonderfulBot.choose\_talon\_step\_2()

Bot gre po vrsti po barvah in si zalaga barve z 1 karto. Če to ni možno oz. jih je premalo kart si začne zalagati barve z 2 kartama. Če še to ni možno potem izbere najbolj vredne karte (ki niso kralji) in si jih založi.

### WonderfulBot.play\_card()

Metoda vrne karto, ki jo določi logika bota. Logika za igranje se razdeli na 3 metode in preverja po temu vrstnemu redu:

* Play\_first(); ko na mizi ni nobene karte in je prvi na vrsti
* Play\_color(); ko je na mizi že neka barva in moram to barvo odigrati.
* Play\_tarot(); ko nimam barve in odigram taroka

### WonderfulBot.play\_first()

Po vrsti gre po sledečih pogojih in če je izpolnjen metoda zaključi v tistem koraku

1. preveri če imam kralja katerga barva še ni bila igrana, če ga imam ga odigraj
2. Če igram igro in imam barvo v kateri igram odigraj najbolj vredno, če pa ne igram potem pa izberem najmanj vredno v barvi, ki se igra
3. poišči če je kakšna barva že bila odigrana in odigraj najnižjo karto v tej barvi (ki ni dama ali kralj)
4. tarokiranje če je le možno po formuli napisano spodaj ALI če ostali igralci nimajo več tarokov
5. odigraj najnižjega taroka
6. random med kartami ki niso kralji ali dame
7. random

Formula za tarokiranje

Če je pogoj izpolnjen potem gre tarokirat kar pomeni da izbere najvišjega taroka.

### WonderfulBot.play\_color()

<https://snipsave.com/user/simonoblak/snippet/PDPH0XXQP9i4Zbqej4/>

### WonderfulBot.play\_tarot()

<https://snipsave.com/user/simonoblak/snippet/JAYcZEPnuxol9iHvTh/>

# **Podatkovna baza**

