**Python - Programmentwurf**

**Team ID:**

**Team Mitglieder (Name + e-mail): Frank Rübenkönig (ruebenkoenig.f-it19@dhbw-ravensburg.de), Sarah Stebich (stebich.sarah-it19@it.dhbw-ravensburg.de), Tobias Spohn (spohn.tobias-it19@it.dhbw-ravensburg.de), Florian Christoph Veit (veit.florian\_c-it19@it.dhbw-ravensburg.de)**

**Git verwendet? (fließt nicht in die Note ein): Ja**

**Buildserver verwendet? (fließt nicht in die Note ein): Ja**

**Verwendete Python Version: 3.8.2**

**Verwendete Pylint Version: 2.4.4**

**Beschreibung des Dateiformats:**

* Wie ist die Datei aufgebaut?

Es wird als Binärdatei gespeichert. Das Modul pickle wird zum Laden und Speichern genutzt. Damit konnten wir eine Einfache und schnelle Umsetzung realisieren. Die Datei ist damit nicht Verschlüsselt. Das hat den Vor- und Nachteil, dass jeder diese Lesen, Schreiben und Nutzen kann.

Die Log Datei wird im txt-Format gespeichert damit Sie mit jedem Editor geöffnet werden kann.

**Beschreibung des Benutzerinterfaces:**

* Wie sieht die Ein- und Ausgabe aus?

Es wird immer ein Header und Spielfeld ausgegeben. Die Eingabe erfolgt immer darunter.

Nach einer Eingabe wird die Anzeige im Terminal gelöscht und neu Ausgegeben.

Im Header stehen die Funktionen: Neues Siel, Speichern, Laden und Spiel Beenden.

Das Spielfeld ist von zwei horizontalen Linien umgeben. Es werden schwarze und weiße Quadrate genutzt um das Spielfeld darzustellen (◻ /◼). Die Bauern sind in den richtigen Farben die richtigen Zeichen (♙/♟). Dieses Zeichenkonstrukt wird von Koordinaten (A…H/1…8) umgeben.

Beim wählen von Figuren werden mögliche Züge und die gewählte Figur in einer anderen Farbe dargestellt.

**Beschreibung der Architektur:**

* Änderungen an der Vorlage

Ordnerstruktur geändert. Diese ist nun eine Realisierung von Modulen.

* Grobe Architektur

Diese wurde in der Architektur.graphml geplant, sowie in den einzelnen Modulen eine graphml Datei liegt.

Es wurde in vier Module unterteilt: User Interface, Computer Gegner, Logik (Spielfeld und Figuren) und Dateiverwaltung.

**Beschreibung des Computergegners (4er Gruppen):**

* Welcher Ansatz wird verfolgt

***Sorry weiß gerade nichts dazu!***

**Log von einem Spiel:**

* Ein Spieldurchlauf dokumentieren

Dateiname:

**Log von den Tests:**

**User Interface:**

**Computer Gegner:**

**Logik:**

**Dateiverwaltung:**

**Bewertung der Testergebnisse:**

* Grund für fehlschlagende Tests

**Code-Coverage Ausgabe:**

**Bewertung der Coverage und Sinnvollheit der Tests:**

* Gewünscht ist eine Coverage von min 75%
  + Wurde dies erreicht? Wenn nein, warum nicht?
* Begründet warum ihr euch sicher seid alles mit Tests abgedeckt zu haben
* Fehlerfälle in den Tests überprüft?

**Bewertung der Fehlersicherheit:**

* Werden Fehleingaben korrekt abgefangen?
  + Nenne 2 Beispiele wo ein Fehler abgefangen wird

**Pylint Ausgabe:**

**Bewertung der Code-Qualität/Lesbarkeit:**

* Begründet noch vorhandene Pylint Warnungen
* Begründet warum euer Code gut lesbar ist

**Bewertungstabelle:**

* Ausgefüllte Bewertungstabelle