Beschlussprotokoll

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Freitag 13.10.2023 | Zeit | 10:10 |
| Ort | TS 01.13 | | |
| Sitzungsleitung | Feuchter Moritz | Protokollführer | Hofer Levin |

Teilnehmende: Moritz Feuchter, Michael Voemel, Julien Wyss, Levin Hofer, Patrick Feisthammel

|  |  |
| --- | --- |
| Inhalte   1. Auffrischung Klassendiagramm 2. Besprechung und Feedback zum Klassendiagramm 3. Testing Konzept 4. Ausblick | Anlagen  Besprechungsprotokoll 13.10.2023 (nicht überarbeitet) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nr. | Thema/Massnahme | Termin | Verantwortlich | Erledigt |
| 1 | Testkonzept weiter ausführen und anpassen für bessere Verständlichkeit und für Zielbringende Optimierung | 18.10.2023 | Moritz Feuchter | ✓ |
| 2 | Codeverbesserungen bezüglich Clean Code regeln | 22.10.2023 | Gruppe |  |
| 3 | Verbesserung der Optische Darstellung des Klassendiagrammes | 22.10.2023 | Gruppe |  |
| 4 | Fertigschreiben des Programmes nach Anforderungen | 27.10.2023 | Gruppe |  |
| 5 | Testen des Programmes nach Selbs festgelegten Testkonzept | 27.10.2023 | Gruppe |  |
| 6 |  |  |  |  |

Sitzungsleitung, Datum, Unterschrift Protokollführer, Datum, Unterschrift

Freihandzeichnungen
Freihandzeichnungen
Freihandzeichnungen
Freihandzeichnungen
Freihandzeichnungen
Freihandzeichnungen
Freihandzeichnungen
Freihandzeichnungen
Freihandzeichnungen
Freihandzeichnungen
Freihandzeichnungen
Freihandzeichnungen


16.10.2023 M.Feuchter \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Die wichtigsten Infos wurden Grau gekennzeichnet.

# 1 Testkonzept weiter ausführen

Bei der Besprechung wurde betont, dass das Testkonzept klar formuliert sein sollte, um zu verstehen, was getestet werden muss und wie die Automatisierung erfolgt. Frontend-Testing sollte vermieden werden, da es Zeit- und Ressourcenaufwand erfordert. Demnach wird das Testkonzept wird überarbeitet, um die Feedbackpunkte zu berücksichtigen, und dann erneut zur Kontrolle und für Verbesserungsvorschläge vorgelegt.

Von uns werden spezifischere Testfälle für die wichtigsten Funktionen und verschiedenen zustände des Programms erstellt, einschließlich der Validierung von Spielzügen und der Überprüfung des Spielstatus. Ebenfalls werden Test für die wichtigsten vom Spieler ausgeführten Zügen erstellt.

# 2 Codeverbesserungen

Uns ist bewusst, dass die Strukturierung von Java-Code nach den Clean Code-Regeln von entscheidender Bedeutung ist, um den Quellcode wartbar, verständlich und effizient zu gestalten. Wir werden deshalb versuchen das untere Punkten so gut wie möglich umzusetzen.

1. Aussagekräftige Namen verwenden

2. Kurze und prägnante Methoden

3. Vermeiden von tiefen Verschachtelungen

4. Eindeutige Verantwortlichkeiten

5. Konsistente Formatierung

6. Kommentare sparsam verwenden

# 3 Verbesserung Klassendiagrammes

Das Klassendiagramm wird optisch überarbeitet, um die Übersichtlichkeit zu verbessern. Es werden einige Interne private Felder nicht im Diagramm dargestellt, es sei denn, sie haben eine wichtige externe Bedeutung oder liefern einen Mehrwert. Des Weiteren wird auf die verschiedenen Abhängigkeiten der einzelnen Klassen und ihrer Darstellung geachtet. Dies soll der Korrektheit sowie der besseren Lesbarkeit dienen.

# 4 Fertigschreiben des Programmes

Das Programm wird gemäss unserer Planung aufgeteilt auf die Gruppe fertig geschrieben.

# 5 Testen des Programmes

Das Programm wird gemäss dem von uns festgelegten Testkonzept welches nach erneuter Abgabe an die Lehrkräfte unter Beachtung des dort erhaltenen Feedbacks, getestet.

Das Ausführen von Tests in einem Java-Programm ist ein kritischer Schritt, um die Qualität und die fehlerfreie Funktionalität unserer Anwendung sicherzustellen.