Arbeitsjournal vom 29.11.2021

|  |  |
| --- | --- |
| Zeit (geplant) | 3h |
| Zeit (tatsächlich) | 3h |
| Geplante Ziele | Mit GitHub arbeiten und Kennenlernen, Projekt Thema nach Anforderungen auswählen. |
| Erreichte Ziele | Beispiel Projekte auf GitHub Hochgeladen, Projekt Thema entschieden (Snake). |
| ausgeführte Arbeiten | * Zwei Projekte als Test auf GitHub hochgeladen mit Git * nach Projekt Themen gesucht und mit den Anforderungen ausgewählt. |
| Probleme | Gab keine Probleme |
| Hilfestellung | <https://github.com/almir12A/> |

Arbeitsjournal vom 6.12.2021

|  |  |
| --- | --- |
| Zeit (geplant) | 3h |
| Zeit (tatsächlich) | 3h |
| Geplante Ziele | Ordnerstruktur auf GitHub hochladen |
| Erreichte Ziele | Manche Dateien auf GitHub hochgeladen |
| ausgeführte Arbeiten | * Ordnerstruktur auf GitHub hochzuladen * Git befehle notiert |
| Probleme | Ich konnte nicht alles hochladen die Befehle haben nicht mehr funktioniert |
| Hilfestellung | <https://github.com/almir12A/> |

Arbeitsjournal vom 13.12.2021

|  |  |
| --- | --- |
| Zeit (geplant) | 3h |
| Zeit (tatsächlich) | 3h |
| Geplante Ziele | Die Probleme von letzten mal zu lösen und UML Klassendiagramm machen und die Klassen dann zu erstellen |
| Erreichte Ziele | Problem noch nicht gelöst, UML Klassendiagramm Fertiggestellt |
| ausgeführte Arbeiten | * Die Probleme mit GitHub konnten noch nicht gelöst werden * UML Klassendiagramm erstellt Verbindungen gemacht |
| Probleme | Mit GitHub |
| Hilfestellung |  |

Arbeitsjournal vom 3.01.2022

|  |  |
| --- | --- |
| Zeit (geplant) | 3h |
| Zeit (tatsächlich) | 3h |
| Geplante Ziele | Die Klassen zu erstellen mit Getter und Setter |
| Erreichte Ziele | Klassen und Getter, Setter wurden erstellt |
| ausgeführte Arbeiten | * Die Klassen erstellt mit dem UML Klassendiagramm * Setter und Getter erstellt |
| Probleme | Mit dem automatischen Erstellen von Getter und Setter aber konnte verbessert werden. |
| Hilfestellung |  |

Arbeitsjournal 4.01.2022

|  |  |
| --- | --- |
| Zeit (geplant) | 3h |
| Zeit (tatsächlich) | 5h |
| Geplante Ziele | Das GitHub Problem lösen mit dem Lehrer, Klasse Graphics das Spielfeld (Apple, Snake) erstellen und state START,RUNNING,END. |
| Erreichte Ziele | Die Probleme mit GitHub Gelöst, Spielfeld erstellt |
| ausgeführte Arbeiten | * Mit dem Lehrer das mit GitHub angeschaut und verbessert * Spielfeld, Snake und Apple erstellt * State Start: man kann eine taste klicken dann erscheint das Spiel * State Running: Das Spiel Läuft * State End: das spiel wird beendet und man sieht sein Score |
| Probleme | Mit JPanel paintComponent(Graphics g) |
| Hilfestellung | <https://www.markdownguide.org/cheat-sheet/>  <https://stackoverflow.com/questions/15544549/how-does-paintcomponent-work> |

Arbeitsjournal 10.01.2022

|  |  |
| --- | --- |
| Zeit (geplant) | 4h |
| Zeit (tatsächlich) | 4,30h |
| Geplante Ziele | Snake Bewegen und Apple essen und wird random respawnt |
| Erreichte Ziele | Snake kann sich bewegen, Apple wird gegessen und wird random respawnt |
| ausgeführte Arbeiten | * Snake kann sich bewegen mit Keylistener * Wenn Snake Apple berührt, soll es verschwinden und Respawnen * Apple spawnt random im Spielfeld |
| Probleme |  |
| Hilfestellung | M226A RobiGame |

Arbeitsjournal 11.01.2022

|  |  |
| --- | --- |
| Zeit (geplant) | 4h |
| Zeit (tatsächlich) | 4h |
| Geplante Ziele | Snake wird grösser, wenn er den Apfel isst, |
| Erreichte Ziele | Snake wird nur einmal grösser |
| ausgeführte Arbeiten | * Wenn Snake den Apfel gegessen hat wird mit body.add(rect) um eins grösser |
| Probleme | Es wird nur einmal grösser |
| Hilfestellung |  |

Arbeitsjournal 16.01.2022

|  |  |
| --- | --- |
| Zeit (geplant) | 3h |
| Zeit (tatsächlich) | 3h |
| Geplante Ziele | Snake wird grösser, wenn er den Apfel isst, |
| Erreichte Ziele | Snake wird immer grösser, wenn er ein Apfel isst, |
| ausgeführte Arbeiten | * Wenn Snake den Apfel gegessen hat, wird mit body.add(rect) grösser * Mit rect.setLocation((w / 2 - 2) \* d, (h / 2) \* d); |
| Probleme | Rect.setLocation |
| Hilfestellung | <https://www.learnjavacoding.com/definitions/snake/>  <https://www.youtube.com/watch?v=91a7ceECNTc> |

Arbeitsjournal 17.01.2022

|  |  |
| --- | --- |
| Zeit (geplant) | 3h |
| Zeit (tatsächlich) | 4h |
| Geplante Ziele | Score, Highscore mit .txt File |
| Erreichte Ziele | Wenn man einen Apfel gegessen hat, steigt der Score. |
| ausgeführte Arbeiten | * Mit s.getBody().size() -3 nimmt man die jetzige Grösse minus der Standard Grösse 3 und das ist dann der Score. * Das Gleiche versucht mit der Highscore wie beim Robigame |
| Probleme | Highscore hat nicht funktioniert |
| Hilfestellung | M226A RobiGame |

Arbeitsjournal 18.01.2022

|  |  |
| --- | --- |
| Zeit (geplant) | 1h |
| Zeit (tatsächlich) | 1h |
| Geplante Ziele | Highscore mit .txt File einbauen |
| Erreichte Ziele | Highscore funktioniert noch nicht |
| ausgeführte Arbeiten | * Versucht den Highscore vom RobiGame im SnakeGame einzubauen |
| Probleme | Probleme mit den Parametern und Highscore wird aufgerufen aber passiert nix |
| Hilfestellung | M226A RobiGame |

Arbeitsjournal 24.01.2022

|  |  |
| --- | --- |
| Zeit (geplant) | 8h |
| Zeit (tatsächlich) | 9h |
| Geplante Ziele | Highscore mit .txt File einbauen, Projekt Dokumentation, Alles auf GitHub hochladen |
| Erreichte Ziele | Highscore funktioniert, Projekt Dokumentation Fertig, Alles auf GitHub hochgeladen |
| ausgeführte Arbeiten | * Highscore wird in einem .txt File gespeichert mit dem Namen, File wird erstellt und gelesen in der GameController Klasse. * Projekt Dokumentation * Alles auf GitHub hochgeladen |
| Probleme | Highscore in der Graphics Klasse hat nicht funktioniert in der GameController hat es Funktioniert |
| Hilfestellung | https://www.youtube.com/watch?v=8gMd0ftWp\_Y |