

## **2. Anforderungen, Projekt, Funktionalität**

### **2 – Hallooo Matematikeeer**

Konzulent:  
**Márton Kovács**

#### **Mitglieder**

Henrietta Domokos	J0DCPT	domoheni@gmail.com
Ábel Borbély	DRP0DT	babel1122@gmail.com
Renátó Hrotkó	OIT6HD	renatohrotko@gmail.com
Vince Pongrácz	MKMDJO	pongrrvin@gmail.com

17.02.2021

## **2. Anforderungen, Projekt, Funktionalität**

### **2.1 Einführung**

#### **2.1.1 Ziel**

Diese Dokument beschäftigt sich mit dem Asteroidenspiel von dem Team Hallooo Mathematikeer. In diesem Dokument kann man über Planung, Implementation usw. lesen.

#### **2.1.2 Fachgebiet**

Dieser Software ist ein Computerspiel, also es lässt sich nicht behaupten, dass es zu einem bestimmten Fachgebiet gehört. Sondern ist der Ziel unsere fachliche Entwicklung und die Unterhaltung der Spieler.

#### **2.1.3 Definitionen, Abkürzungen**

*gs.: grundsätzlich*

*wi.: wichtig*

*opt.: optionell*

*Besch.: Beschreibung (Aufgabenbeschreibung)*

*TDR: Team Defined Requirement (Team definierte Anforderung)*

#### **2.1.4 Bezug**

*Vorige Hausaufgaben aus Softwaretechnologie*

*Aufgabebeschreibung: <https://www.iit.bme.hu/file/11582/feladat>*

#### **2.1.5 Zusammenfassung**

Die Folgenden sind in diesem Dokument zu finden:

- Überblick: Das stellt der ganze Projekt in einem kurzen Form vor.
- Anforderungen: Diese Anforderungen werden später in Betracht genommen werden und beim Softwareentwurf benutzt.
- Wichtige Use-Case Diagramme: Die wichtigste Diagramme werden aufgelistet und beschreibt.
- Wörterbuch: Die Erklärung/Definition der solchen nicht alltäglichen Wörter, die in diesem Dokument und in dem Spiel vorkommt.
- Projekt Plan: Die realisierende Aufgaben und Termine werden aufgelistet.
- Tagebuch: Die fertige Aufgaben werden hier angegeben und auch die benötigte Zeit.

## **2.2 Überblick**

### **2.2.1 Allgemeiner Überblick**

Die Grundarchitektur des Programms besteht aus zwei Hauptelementen. Die Asteroiden, und die Dinge die sich auf ihnen befinden. Diese Dingen sind die Raumschiffe der Siedler, die von den Spielern gesteuert sind, autonome Roboter, die sich selbst steuern, und Teleporterpaare, die Verkehr zwischen nicht benachbarten Asteroiden ermöglichen.

### **2.2.2 Funktionen**

Die Software ist ein Computerspiel, in dem Spieler versuchen, ein gemeinsames Ziel zu erreichen. Das Spiel findet in einem Asteroidenfeld statt. Die Spieler übernehmen die Rolle von Siedlern, die mit den im Kern der Asteroiden enthaltenen Ressourcen eine Basis aufbauen wollen. Es gibt Elemente, die den Spielern helfen können, dieses Ziel zu erreichen, und es gibt einige, die ihren Fortschritt behindern oder sie sogar töten.

Ziel des Spieles: Die Spieler müssen Ressourcen sammeln. Die Spieler gewinnen, wenn mindestens drei Einheiten jedes Materials auf einem einzelnen Asteroiden gesammelt werden.

Reisen durch das Spielfeld: Die Spieler starten auf einem HomeAsteroid, worauf sie die Basis aufbauen müssen. In einem Schritt können Spieler von einem Asteroiden zum anderen Reisen, aber mit der wichtigen Einschränkung, dass sie benachbart sind. Zu einem Asteroiden kann theoretisch von einigen bis Hunderte von Nachbarn gehören. Siehe auch: Teleporterpaare.

Ressourcen: Im Kern der meisten Asteroiden finden Spieler Ressourcen. Das innere von Asteroiden ist homogen, so dass Sie nur eine Einheit von einem Material in einem einzelnen Asteroiden finden können. Es gibt auch hohle Asteroiden, die überhaupt nichts enthalten. Die möglichen Materialien sind Eis, Eisen, Kohle und Uran.

Bohren und Bergbau nach Ressourcen: Um Ressourcen sammeln zu können, müssen die Spieler die äußere Schichten der Asteroiden die den Kern bedecken durchbohren. Nur dann können sie die Ressource abbauen, wenn überhaupt eine vorhanden ist. Ein Siedler kann höchstens 10 Ressourceneinheiten halten. Wenn der Spieler es wünscht, kann er eine Ressourceneinheit in einem hohlen, gebohrten Asteroiden aufbewahren.

Autonome Roboter bauen: Die Spieler können die gesammelten Ressourcen verwenden, um autonome Roboter zu bauen, die gemäß den zuvor genannten Regeln reisen können. Ansonsten können sie die äußeren Schichten eines Asteroiden bohren, aber nicht die Ressourcen darin abbauen. Roboter können aus jeweils einer Einheit Eisen, Kohle und Uran gebaut werden.

Teleporterpaare: 2 Einheiten Eisen und 1 Einheit Eis und Uran sind erforderlich, um ein Paar Teleporter zu bauen. Spieler können Teleporter im Asteroidenfeld platzieren. Wenn beide Teleporter platziert sind, können Spieler und Roboter zwischen ihnen reisen. Ein Spieler kann höchstens ein Paar Teleporter halten.

Radioaktive Ressourcen und die Sonne: Die Spieler müssen vorsichtig mit Asteroiden sein, die der Sonne nahe sind. Uran ist offensichtlich radioaktiv. Ein Asteroid mit radioaktivem Kern explodiert, wenn er sich in der Nähe der Sonne befindet, und seine Außenseite ist vollständig durchbohrt, wodurch die Siedler auf diesem Asteroiden getötet werden. Roboter überleben die radioaktive Explosion und landen auf einem benachbarten Asteroiden.

Solarflairs: Von Zeit zu Zeit treten Sonnenflair auf, die den Asteroidengürtel erreichen. Dies wirkt sich sowohl auf Roboter als auch auf Spieler aus und kann nur überlebt werden, wenn sie sich auf einem vollständig gebohrten Asteroiden mit hohlen Innenseiten befinden (sie verstecken sich).

Spiel vorbei: Das Spiel ist vorbei und die Spieler verlieren, wenn jeder Siedler stirbt.

### 2.2.3 Benutzern

Die Benutzer sollen keine extra, oder vorausgesetzte Kenntnisse über das Spiel haben, unser Ziel ist ein selbstverständige Benutzerschnittstelle und ein genüssliche Spiel am Ende der Entwicklungsprozess herstellen.

### 2.2.4 Begrenzungen

Das Spiel soll die definierte Anforderungen, und Spielscenarios realisieren. Es muss auch stabil sein, muss es so funktionieren, wie es erwartet werden kann. z.B: eine unerwartete Stillstand ohne Grund, oder ohne additionelle Information ist nicht erlaubt.

### 2.2.5 Voraussetzungen, Beziehungen

*Der Benutzer soll nicht vollständig blöd sein, z.B: wenn eine Taste gedrückt werden soll, dann sei diese Taste nicht das Ein- und Ausschaltertaste der Maschine.*

*Aufgabebeschreibung: <https://www.iit.bme.hu/file/11582/feladat>*

*Unsere vorige Programmierung 3, und Softwaretechnologie Notizen, und aus diesem Fach wurde benutzt.*

## 2.3 Anforderungen

### 2.3.1 Funktionale Anforderungen

Prioritäten:

- *gs*: grundsätzlich
- *wi*: wichtig
- *opt*: optionell

Anforderungen aus:

- *Besch.*: Beschreibung (Aufgabenbeschreibung)
- *TDR*: Team Defined Requirement (Team definierte Anforderung)

ID	Beschreibung	Kontrolle	Priorität	Quelle	Use-case	Komment
1.01	Die Spieler steuern die Siedler	Präsentation	gs	Besch.	Alle use-case	-
1.02	Die Spieler und Roboter können von einem zu einen anderen Asteroid bewegen.	Präsentation	gs	Besch.	Move Settler	-
1.03	Die Spieler und Roboter können auf einem Asteroid bohren.	Präsentation	gs	Besch.	Drill Asteroid	-
1.04	Die Roboter helfen die Spieler.	Auswertung	wi	Besch.	-	-
1.05	Spieler können Roboter erzeugen	Präsentation	gs	Besch.	Create Robot	-
1.06	Spieler können JumpGate-en herstellen.	Präsentation	gs	Besch.	Manage JumpGate	-
1.07	Asteroiden haben Mantelgröße, die verschieden sein können.	Auswertung	wi	Besch.	-	-
1.08	Ein Rohstoff befindet sich in den Kern einer Asteroid	Präsentation	gs	Besch.	-	-
1.09	Existieren gefährliche Asteroiden, die explodieren kann.	Präsentation	gs	Besch.	-	-
1.10	Existiert mindestens eine	Präsentation	gs	Besch.	-	-

	<i>Asteroid, und ein Asteroidenfeld</i>					
1.11	<i>Es gibt mehrere Rohstoffe, die sind WasserEis, Eisen, Kohle, und Uran.</i>	<i>Präsentation</i>	<i>gs</i>	<i>Besch., TDR</i>	-	-
1.12	<i>Ein Asteroid hat ein Kern, die genau eine Rohstoff enthielt.</i>	<i>Präsentation</i>	<i>gs</i>	<i>Besch.</i>	-	-
1.13	<i>Gibt's Asteroiden, die keine Rohstoff enthalten, sondern Hohlraum.</i>	<i>Präsentation</i>	<i>gs</i>	<i>Besch.</i>	-	-
1.14	<i>In einem Schritt kann ein Astronaut (Spieler) nur eine Tätigkeit durchführen</i>	<i>Auswertung</i>	<i>wi</i>	<i>Besch.</i>	<i>Alle use-case</i>	-
1.15	<i>Andere mögliche Tätigkeiten einer Spieler sind: bewegen, bohren, fördern, JumpGate auf- und abbauen, Resource ablegen</i>	<i>Präsentation</i>	<i>gs</i>	<i>Besch.</i>	<i>Move Settler, Drill Asteroid, Mine Asteroid, Manage JumpGate, Create Robot</i>	-
1.16	<i>Möglichkeiten sind noch die Nachbarnatseroid en auslisten.</i>	<i>Präsentation</i>	<i>wi</i>	<i>TDR</i>	-	<i>Nützlich kann sein beim Bewegen.</i>
1.17	<i>Spieler kann eine bestimmte Menge der Ressourcen mitnehmen/transp ortieren.</i>	<i>Präsentation</i>	<i>gs</i>	<i>Besch.</i>	<i>Move Settler</i>	<i>Diese Menge wird wahrsc hei nlich 10. (integriert im Move Settler use-case)</i>
1.18	<i>Jede Asteroid hat beliebig viele Nachbarn.</i>	<i>Präsentation</i>	<i>wi</i>	<i>Besch., TDR</i>	-	<i>Es kann mehrere Hunderte sein, aber mindestens 1.</i>

1.19	Beim Bohren wird das Loch mit einer Einheit tiefer.	Auswertung	gs	Besch.	Drill Asteroid	Loch wird so modelliert, dass die Manteldicke kleiner wird
1.20	Fördern ist dann und nur dann möglich, wenn das Loch fertig ist	Auswertung	gs	Besch.	Mine Asteroid	Loch ist fertig, wenn der Mantel ist völlig durchbohrt.
1.21	Asteroiden, JumpGate-n, und Sonne hat Position im universum	Auswertung	opt	TDR	-	-
1.22	Asteroid kann in der Sonnennähe, oder fern von der Sonne liegen.	Auswertung	wi	Besch.	-	-
1.23	Wenn in der Sonnennähe liegende Asteroid radioaktive Rohstoff gebohrt wurde, Asteroid explodiert.	Auswertung/Päsentation	gs	Besch.	Mine Asteroid	-
1.24	Gibt's Solarflair, das die Astronauten, und Roboter tötet, wenn sie nicht im Hohlraum einer Asteroid sind	Präsentation	gs	Besch.	-	-
1.25	Roboter sind von dem Spiel kontrolliert.	Präsentation	wi	Besch.	-	-
1.26	Roboter überleben die radioaktive Explosion, aber sie werden auf einer Nachbarasteroid geworfen.	Präsentation	gs	Besch.	-	-
1.27	Die Spieler haben gewonnen, wenn aus alle Rohstoffe mindestens 3	Präsentation	gs	Besch.	-	-

	<i>Einheit gefördert, und diese Rohstoffe auf einer Asteroid transportiert wurde.</i>					
1.28	<i>Das Spiel hat zwei Szenarien zum Spielende: Gewinnen, und Verloren</i>	<i>Präsentation</i>	<i>gs</i>	<i>Besch.</i>	<i>-</i>	<i>-</i>
1.29	<i>Die Spieler haben verloren, wenn die allen tot sind.</i>	<i>Präsentation</i>	<i>gs</i>	<i>Besch.</i>	<i>-</i>	<i>-</i>
1.30	<i>Um eine Robot herzustellen muss man bestimmte Ressourcen bezahlen.</i>	<i>Präsentation</i>	<i>wi</i>	<i>Besch.</i>	<i>Create Robot</i>	<i>1 Stahl, 1 Kohl, 1 Uran</i>
1.31	<i>Mehrere Spieler können spielen.</i>	<i>Präsentation</i>	<i>wi</i>	<i>TDR</i>	<i>-</i>	<i>1 und mehrere</i>
1.32	<i>Eine Asteroid kann eine Spieler oder Robot von einem Solarflair schützen, wenn es Hohlraum hat.</i>	<i>Präsentation</i>	<i>wi</i>	<i>TDR</i>	<i>-</i>	<i>-</i>
1.33	<i>Wenn der Astronaut bewegt sich, er kann aus den Asteroiden wählen, dass sie auf welchem Asteroid fahren möchte.</i>	<i>Präsentation</i>	<i>wi</i>	<i>TDR</i>	<i>Move Settler</i>	<i>-</i>
1.34	<i>Ein Astronaut kann aus dem Spiel austreten</i>	<i>Präsentation</i>	<i>opt</i>	<i>TDR</i>	<i>Leave Game</i>	<i>-</i>
1.35	<i>Beim Fördern wird das ganze Rohstoffmenge aufgefordert (eine Einheit aus dem Rohstoff)</i>	<i>Auswertung</i>	<i>wi</i>	<i>Besch.</i>	<i>Mine Asteroid</i>	<i>-</i>
1.36	<i>Das Spiel soll die Astronauten über einem Solarflair vor dem Solarflair irgendwie berichten.</i>	<i>Präsentation</i>	<i>opt</i>	<i>TDR</i>	<i>-</i>	<i>-</i>



1.37	Solarflairs können den Asteroidzone erreichen	Präsentation	wi	Besch.	-	Nicht immer.
1.38	Roboter können nicht fördern.	Auswertung	wi	Besch.	-	-
1.39	Nur eine Sonne existiert in dem Spiel	Präsentation	wi	TDR	-	wichtig!
1.41	Ein Spieler kann maximal 2 JumpGate mitnehmen	Präsentation	gs	Besch.	-	-
1.42	Um eine JumpGate herzustellen muss man bestimmte Ressourcen bezahlen.	Präsentation	wi	Besch.	Manage JumpGate	2 Stahl, 1 WasserEis und 1 Uran.
1.43	Gibt's ein HomeAsteroid, die größere Speicherkapazität hat.	Präsentation	wi	TDR	-	unendlich e Speicherkapazität (wahrscheinlich)

### 2.3.2 Anforderungen über Ressourcen

ID	Beschreibung	Kontrolle	Priorität	Quelle	Komment
2.01	Programm wird auf Java geschrieben	Auswertung	gs	Besch.	-
2.02	Existenz von JVM	Auswertung	gs	Besch./TDR	
2.03	Windows 10, MAC OS X Betriebssysteme empfohlen	Präsentation	opt	TDR	Diese benutzen wir in der Gruppe
2.04	Maus, Tastatur und Bildschirm angeschlossen	Präsentation	wi	TDR	
2.05	Berechtigungen über Dateiverwaltung	Präsentation	wi	TDR	

### 2.3.3 Anforderungen über Übergabe

ID	Beschreibung	Kontrolle	Priorität	Quelle	Komment
----	--------------	-----------	-----------	--------	---------

3.01	<i>Grundanforderungen sind erfüllt</i>	<i>Auswertung</i>	<i>gr</i>	<i>TDR</i>	-
3.02	<i>Notwendige Softwares sind installiert</i>	<i>Auswertung</i>	<i>gr</i>	<i>TDR</i>	-

### 2.3.4 Andere nicht funktionale Anforderungen

<b>ID</b>	<b>Beschreibung</b>	<b>Kontrolle</b>	<b>Priorität</b>	<b>Quelle</b>	<b>Komment</b>
4.01	<i>Ausführbare Datei mit kleinen Größe</i>	<i>Auswertung</i>	<i>opt</i>	<i>TDR</i>	<i>Wegen Portabilität</i>
4.02	<i>Die Software soll fähig sein mehrere Spieler behandeln</i>	<i>Präsentation</i>	<i>wi</i>	<i>Besch. und TDR</i>	-
4.03	<i>Die Software muss auf Windows und MAC OSX laufen</i>	<i>Auswertung</i>	<i>wi</i>	<i>TDR</i>	<i>Unser Team arbeitet auf Win und Mac.</i>

## 2.4 Wesentliche use-case Diagramme

### 2.4.1 Use-case Beschreibungen

<b>Name des Use-case</b>	<b>Start game</b>
<b>Kurzbeschreibung</b>	Das Spiel wird gestartet.
<b>Aktoren</b>	Player
<b>Drehbuch</b>	<ol style="list-style-type: none"> <li>1. Benutzer drückt START GAME Taste.</li> <li>2. Spieler wartet</li> <li>3. Spiel fängt an.</li> </ol>

<b>Name des Use-case</b>	<b>Leave game</b>
<b>Kurzbeschreibung</b>	Der Spieler tritt aus dem Spiel. Das Spiel endet.
<b>Aktoren</b>	Player
<b>Drehbuch</b>	<ol style="list-style-type: none"> <li>1. Benutzer drückt LEAVE GAME Taste.</li> <li>2. Das Spieler beendet das Spiel.</li> </ol>

<b>Name des Use-case</b>	<b>Pause game</b>
<b>Kurzbeschreibung</b>	Der Spieler kann eine Spielpause einlegen. Das Spiel stoppt, die Ergebnisse bleiben erhalten, bis der Spieler die Pause unterbricht.
<b>Aktoren</b>	Player
<b>Drehbuch</b>	<ol style="list-style-type: none"> <li>1. Benutzer drückt PAUSE GAME Taste.</li> <li>2. Das Spiel stoppt.</li> </ol>

<b>Name des Use-case</b>	<b>Move settler</b>
<b>Kurzbeschreibung</b>	Der Spieler leitet einen Siedler in dem Solarsystem.
<b>Aktoren</b>	Player
<b>Drehbuch</b>	Der Siedler bewegt auf dem nächsten benachbarten Asteride.

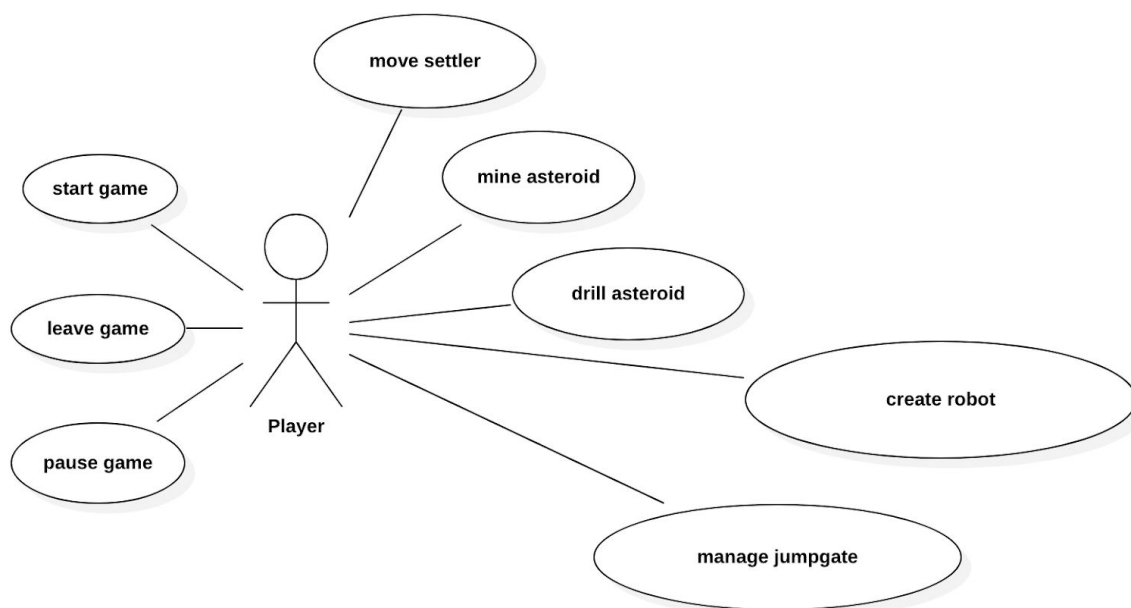
<b>Name des Use-case</b>	<b>Mine asteroid</b>
<b>Kurzbeschreibung</b>	Der Siedler fördert den Rohstoff.
<b>Aktoren</b>	Player
<b>Drehbuch</b>	Der Siedler fördert der Ressource aus dem Kern der Asteroide.

<b>Name des Use-case</b>	<b>Drill asteroid</b>
<b>Kurzbeschreibung</b>	Der Spieler bohrt den Mantel der Asteroide.
<b>Aktoren</b>	Player
<b>Drehbuch</b>	Der Spieler während des Bohres macht den Mantel der Asteroide 1 Einheit kleiner.

<b>Name des Use-case</b>	<b>Manage jumpgate</b>
<b>Kurzbeschreibung</b>	Der Spieler handhabt das Teleporttor.
<b>Aktoren</b>	Player
<b>Drehbuch</b>	Mit zwei Einheiten Eisen, einer Einheit Wassereis und einer Einheit Uran kann der Spieler ein Paar Teleport Tore herstellen. Jedes Tor kann später vom Spieler in der Nähe des gerade besuchten Asteroiden gestartet werden. Die beiden Mitglieder des Tor Paares sind verbunden und betreten eines, der Spieler befindet sich im anderen. Frisch hergestellte Tore können vom Spieler getragen werden, ein Spieler kann jedoch maximal zwei Tore gleichzeitig haben..

<b>Name des Use-case</b>	<b>Create robot</b>
<b>Kurzbeschreibung</b>	Der Spieler baut einen Roboter
<b>Aktoren</b>	Player
<b>Drehbuch</b>	Siedler können mit einer Einheit Eisen, einer Einheit Kohle und einer Einheit Uran einen autonomen Roboter schaffen, der durch künstliche Intelligenz gesteuert wird. Spieler können die Roboter nicht kontrollieren.

## 2.4.2 Use-case diagram



## 2.5 Wörterbuch

**Asteroide:** Auf einem Besetztes Feld liegende Element, aus denen der Spieler Rohstoffe fördern kann.

**Home Asteroide / Basis Asteroide:** Ein solcher Asteroide auf den die Rohstoffe gesammelt werden um dort das endgültige Basis aufzubauen.

**Basis:** Um das Spiel gewinnen zu können muss es aufgebaut werden aus den bestimmten Rohstoffen.

**Besetztes Feld:** Ein solcher Einheit auf der Spielfeld, auf denen eine Astroide ist.

**Bewegen:** In einem Schritt kann ein Spieler dieser Tätigkeit machen. In diesem Fall bewegt er sich auf den benachbarten Astroide.

**Bohren:** In einem Schritt kann ein Spieler dieser Tätigkeit machen. In diesem Fall wird die Manteldicke der Astroide mit 1 verringert. Falls es 0 ist dann kann der Rohstoff gefördert werden.

**Einheit:** Für die Rohstoffe bedeutet, wie viel aus ihnen zur Verfügung steht.

**Explosion:** Falls es radioaktiver Rohstoff in Sonnennah gefördert wird wird es explodieren und den Spieler töten.

**Fördern:** In einem Schritt kann ein Spieler dieser Tätigkeit machen. In diesem Fall wird aus der Kern der Astroide der dort liegende Rohstoff gefördert.

**Gewinnen:** Falls die Spieler aus allen Rohstoffen 3 Einheiten gesammelt haben dann haben sie gewonnen.

**Kern:** In der Mitte der Astroide ist der Kern aus dem Rohstoff gefördert werden kann. Es kann sogar leer oder radioaktiv sein.

**Leeres Feld:** Ein solcher Einheit auf der Spielfeld, auf denen keine Astroide oder andere Sache ist.

**Manteldicke:** Es zeigt wie dick der Mantel der Astroide ist. Beim Bohren wird es mit 1 verringert.

**Nachbarn:** Die Asteroide haben nachbarn in bestimmten Radius.

**Programm:** Das fertig gemachte und entwickelte Endergebnis.

**Radioaktiv:** Ein Rohstoff kann radioaktiv sein. In diesem Fall wenn die Astroide in Sonnennah ist dann wird es explodieren.

**Raumschiff:** Eigentlich äquivalent mit dem Spieler. Es hat keine spezielle Bedeutung.

**Robot:** Ein Robot mit AI kann bewegen, bohren und wird nicht explodieren in Sonnennahe.

**Robot bauen:** In einem Schritt kann ein Spieler dieser Tätigkeit machen. In diesem Fall kann er aus den bestimmten Rohstoffen ein Robot mit AI bauen.

**Rohstoff:** Es befindet sich in dem Kern der Asteroiden. Aus denen können Roboter, Portals, Basis gebaut werden.

**Schritt:** In einem Schritt kann der Spieler Tätigkeiten durchführen. zB: Fördern, Bewegen, Bohren, Portal und Roboter bauen.

**Siedler:** siehe Spieler.

**Sonnenflair/Solarflair:** ein Sturm von der Sonne und kann die Spieler töten falls sie nicht in einem leeren Astroide sind.

**Sonnennah:** Es bedeutet dass die Spieler nah zu der Sonne sind und falls ein radioaktiver Rohstoff aus ihnen gefördert wird dann wird es explodieren.

**Spieler:** Der Person der den Spiel spielt. Die Siedler leitet und Tätigkeiten durchführt.

**Spielfeld :** Den ganzen Feld auf denen der Spiel passiert. Auf ihm sind Spieler, Asteroiden, Roboter, Portals.

**Sterben:** Falls der Spieler explodiert oder ungeschützt von einem Sonnenflair ist dann wird er sterben und nicht mehr spielen.

**Teleport Portal:** Ein Spieler kann max 2 bei ihm haben. Aber kann mehrere Portal Paare haben. Dadurch kann man teleportieren. Es kann der Spieler für bestimmte Rohstoffe bauen.

**Teleport Portal bauen:** In einem Schritt kann ein Spieler dieser Tätigkeit machen. In diesem Fall wird ein Teleport Portal gebaut. Siehe Teleport Portal.

**Töten:** Ein Sonnenflair oder eine Explosion kann die Spieler vernichten/töten also sie können das Spiel nicht fortsetzen.

**Verlieren:** Falls alle Spieler gestorben sind dann ist der Spiel vorbei und sie haben verloren.

## 2.6 Projekt Plan

### Schritte und ihre Abgabetermine

02.10. Informationsvortrag  
02.17. Anforderungen, Projekt, Funktionalität  
02.24. Analyse Modell 1.  
03.03. Analyse Modell 2.  
03.10. Skeleton planen  
03.17. Skeleton  
03.24. Konzeption von Prototyp  
03.31. Detaillierte Pläne  
04.07. Frühling Pause  
04.14. Prototyp  
04.21. Spezifikation von GUI  
04.28. GUI erstellen  
05.05. GUI und Zusammenfassung  
05.12. Buffer, Wiederholen

### Verantwortungen

Ábel Borbély	Dokumentation, Kodierung, Testen des Kodes
Henrietta Domokos	Dokumentation, Kodierung, UML (Use-Case), Drive, PrettyPrinter
Renátó Hrotkó	Dokumentation, Kodierung, Graphische Oberfläche,
Vince Pongrácz	Dokumentation, Kodierung, UML (Class), Repository

### Techniken

**Google Docs / Google Drive** - Es ist verwendet um das Dokumentation zu schreiben. Alle Dokumente sind auf einem Google Drive geteilt zu denen alle zugreifen können. Alle können hier gleichzeitig schreiben und arbeiten und die vorherige Versionen sind auch zurückzustellen.

**Slack** - Es benutzen wir um das Kommunikation der Mitglieder zu schaffen. Stetige Kommunikation geschieht hier fast jeden Tag wann wir bestimmte Problemen und ihre Lösungsmethoden besprechen und versuchen immer sie Beste zu wählen. Die 2 Gruppen der Deutschsprachigen Ausbildung haben 2 unterschiedliche Channels um dort arbeiten zu können.

**Discord** - 2-mal pro Woche haben Sitzungen wann wir besprechen wer was machen wird. Wir versuchen alle Aufgaben gerecht verteilen auch gemäß der einzelnen Ansprüche. Auf unserem Discord Server können wir alles und schnell besprechen und die Dokumente verteilen wir in Slack oder in Drive.

**Git** - Zum Versionskontroll werden wir Git benutzen. Auf GitHub haben wir einen geteilten Repository wohin wir alle Dateien hochladen werden. Da für eine Team sehr wichtig die Zusammenarbeit ist deswegen wenn etwas schief gegangen ist zB in dem Code dann mit Git können wir das einfach zurückstellen und dann die Fehler korrigieren und dann fortsetzen.

**IntelliJ** - das Kodieren passiert in dieser Umgebung. Das haben wir in den letzten Semester benutzt und läuft auf Mac OSX und Windows. und mit Git kann das leicht zusammengebunden werden.

**Star UML** - UML Use-Case-, Klassen-, Sequenzdiagramme herstellen. Vorher haben wir WhiteStar UML benutzt aber das war schlecht zu benutzen und war auch sehr alt. Deswegen benutzen wir Star UML weil es leichter zu benutzen ist und eine viel bessere Benutzeroberfläche hat.

**Adobe Photoshop** - Um die Bilder des Projektes herzustellen. Bilder über Asteroide, Rohstoffe, Raumschiff, die Felder, Basis usw. werden in Photoshop hergestellt.



## 2.7 Tagebuch

Anfang	Zeitdauer	Teilnehmer	Beschreibung
2021.02.14. 17:00	5 Stunde	Borbély Domokos Hrotkó Pongrácz	Anfangsmeeting: Aufgaben durchlesen, verteilen und interpretieren, offene Fragen besprechen. Organisationale Fragen beantworten. (Git, Repository, Planungs- und Entwicklungssoftware)
2021.02.14	cirka 3 Stunde	Domokos	Tätigkeit: Ziel aufschreiben, Vorige Diagrammen durschauen, Use-case herstellen und beschreiben
2021.02.15.: <ul style="list-style-type: none"> <li>• 8:30-10:00,</li> <li>• 16:45-19:00,</li> <li>• 22:00-23:50</li> </ul>	cirka. 5,5 Stunde	Pongrácz	Tätigkeit: Vorige Diagrammen durschauen, neue UML Klassendiagramm Modell herstellen, Anforderungen klar machen und auf schreiben.
2021.02.15.	insg. 5 Stunde	Hrotkó	Einführung, Projektplan, Wörterbuch schreiben und übersetzen. Besprechen von UML modelle. Gedanken machen über die Realisation des Spiels.
2021.02.16. 17:00	3 Stunden	Borbély	Allgemeiner Überblick, Funktionen umschreiben
2021.02.16. 20:00	1 Stunde	Hrotkó Pongrácz Domokos Borbély	Besprechen und Überprüfen der Aufgabe, letzte Berührungen
2021.02.19 21:10	cirka 0.5 Stunde	Pongrácz	Planung (UML Class), und Fehlerkorrekturen
2021.02.16.	0.5 Stunde	Domokos	Use-case finalisieren
2021.02.16. 21:15	0.5 Stunde	Hrotkó	Beenden der ausgebliebene Teile des Dokumentes und Übersetzen.