

# Asteroid Mining

Team name: Pied Pipers

Title: Prototype Program

Supervisor:

Dr. Balla Katalin

## Members:

Abdelrahman Desoki	DOHDZF	adesoki@edu.bme.hu
Neda Radonjic	NQC17Y	neda.radonjic@edu.bme.hu
Tushig Bat-Erdene	QBI3JH	bat-erdene.tushig@edu.bme.hu
Chaitanya Arora	BWTFX8	chaitanya.arora@edu.bme.hu
Janibyek Bolatkhan	BOI6FK	janibyekbolatkhan@edu.bme.hu
Kasay Ito	LX5RFB	kasay.ito@edu.bme.hu

April 25, 2022

## 0- Changes made since the last submission

Some of the test cases described in Concept of the Prototype were not executed, as they are not applicable to the current state of the program-they rely on the GUI part, and could not be tested at the moment.

These are

- Showing the GUI
- Decreasing the health of the Settler (we decided against including a health parameter)
- Visibility of the SunStorm
- Carrying the resources

## 10.1 Deployment guide

### 10.1.1 List of files

Filename	Size	Date	Content
Asteriod.java	2.53KB	2022.04.24	The class contains the GUI and major attributes of the class Asteroid like depth, hollow, distancefromSun etc, and some functions required in rendering the file, getting resources, deepen hole etc which covers most of the use case related to asteroid
Carbon.java	160 Bytes	2022.04.24	Carbon class is a child class of resource
Direction.java	101 Bytes	2022.04.24	The direction is an enum containing the movements up. Down, left right, which is

			used to handle the movement of the spaceship
Game.java	4.85 KB	2022.04.24	The game is one of our main classes, which is initiating the game by beginning the threads when the game begins and renders all objects like asteroids, spaceships etc, which need to be displayed. It contains functions like render, tick and run which control the main structure of the game.
GameObject.java	296 Bytes	2022.04.24	Besides having the normal getter ,setter. This function is used to change the movements of the objects like asteroids and spaceship and check for collisions
Handler.java	1.26KB	2022.04.24	Handler class is for getting neighbor place, adding and removing objects, getting settler, and checking if the asteroid is explosive.
ID.java	230 Bytes	2022.04.24	Enumeration of Settler, Robot, Asteroid, RadioActiveAsteroid, TeleportationGate, SunStorm, Iron, Carbon, Uranium, and WaterIce.

Iron.java	152 Bytes	2022.04.24	Iron class is a child class of resources.
Place.java	1.22KB	2022.04.24	The place class is superclass of
RadioactiveAsteriod.java	725 Bytes	2022.04.24	RadioActiveAsteroid is a superclass of Asteroid with explode method.
Resources.java	196 Bytes	2022.04.24	Class defines the type of resources.
Settler.java	1013 Bytes	2022.03.26	The settler class is the GUI of the settler and will involve the movements and basic rendering.
Robot.java	352 Bytes	2022.04.24	The robot class is the graphical user interface of the robot which involves the movements, basic rendering, and it can get damage.
Settler.java	7.21KB	2022.04.24	The settler class is the GUI of the settler and will involve the movements and basic rendering.
Spaceship.java	1.57 KB	2022.04.24	This class is responsible for initializing spaceship with its

			capacity and current inventory.
Sunstorm.java	957 Bytes	2022.04.24	SunStorm class is GUI implementation sunstorm itself.
Teleportationgate.java	399 Bytes	2022.04.24	The TeleportationGate class is the graphical user interface of the gates.
Uranium.java	164 Bytes	2022.04.24	Uranium class is a child class of resource
Visitor.java	1.70KB	2022.04.24	Visitor class extends the game object and deals with the what things can be done when a settler visit the asteroid and similar functions like drill, hide etc.
WaterIce.java	307 Bytes	2022.04.24	WaterIce class is a child class of resource. It differs with sublime() method from other resources.

### 10.1.2 Compilation

In order to do the successful compilation of the code, one can clone the project or download the zip file and run it in IntelliJ IDE.

To execute the program we need to run the main function present in the Game.java file. Namely, the entry point is this function.

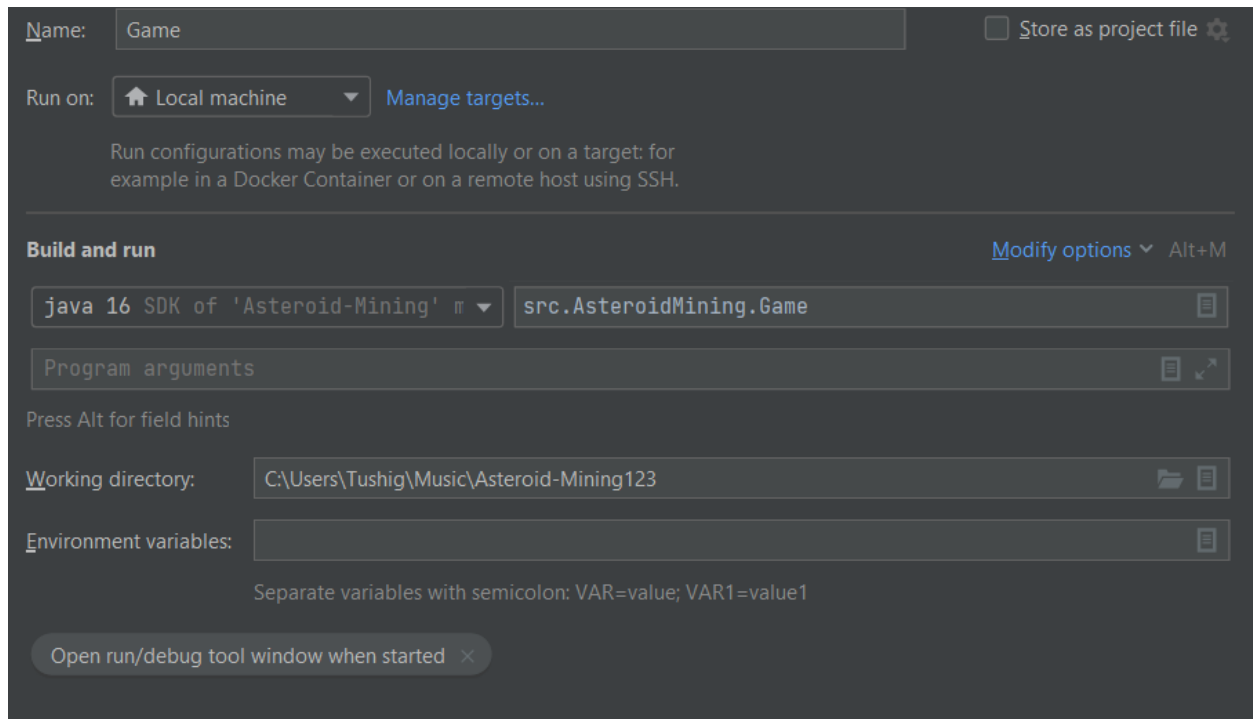
The link to github repository:

<https://github.com/jackdotb/Asteroid-Mining>

### 10.1.3 Run

There are no particular requirements except the fact that jdk should be installed in the users IDE and should preferably use intellij IDE. In order to compile the project successfully, “Game” class is needed to configured in “Run/Debug Configuration” section of the IntelliJ IDE. Especially, some of test cases are suggested that they should be checked in debugging mode with breakpoints.

Run/Debug Configuration:



## 10.2 Test protocols

### 10.2.1 Starting the game

<b>Name of the tester</b>	Neda Radonjic
<b>Date &amp; time of test</b>	24/04/2022 22:42

### 10.2.2 Build a space station

<b>Name of the tester</b>	Neda Radonjic
<b>Date &amp; time of test</b>	24/04/2022 22:45

### 10.2.3 Settler traveling

<b>Name of the tester</b>	Neda Radonjic
<b>Date &amp; time of test</b>	24/04/2022 22:45

### 10.2.4 Settler drilling

<b>Name of the tester</b>	Chaitanya Arora
<b>Date &amp; time of test</b>	24/04/2022 22:43

### 10.2.5 Settler mining

<b>Name of the tester</b>	Chaitanya Arora
<b>Date &amp; time of test</b>	24/04/2022 22:43

### 10.2.6 Filling the asteroid

<b>Name of the tester</b>	Chaitanya Arora
<b>Date &amp; time of test</b>	24/04/2022 22:44

### 10.2.6 Is asteroid in perihelion (asteroid explosion)

<b>Name of the tester</b>	Neda Radonjic
<b>Date &amp; time of test</b>	24/04/2022 23:08

### 10.2.7 Building the robot

<b>Name of the tester</b>	Neda Radonjic
<b>Date &amp; time of test</b>	24/04/2022 22:52

### 10.2.8 Building the teleportation gate

<b>Name of the tester</b>	Neda Radonjic
<b>Date &amp; time of test</b>	24/04/2022 22:52

### 10.2.9 Deploying the teleportation gate

<b>Name of the tester</b>	Neda Radonjic
---------------------------	---------------



<b>Date &amp; time of test</b>	24/04/2022 23:03
--------------------------------	------------------

### 10.2.10 Hiding the settler

<b>Name of the tester</b>	Chaitanya Arora
<b>Date &amp; time of test</b>	24/04/2022 23:03

### 10.2.12 Death of the settler

<b>Name of the tester</b>	Neda Radonjic
<b>Date &amp; time of test</b>	24/04/2022 23:06

### 10.2.13. Checking inventory

<b>Name of the tester</b>	Chaitanya Arora
<b>Date &amp; time of test</b>	24/04/2022 23:12

### 10.2.14 End game

<b>Name of the tester</b>	Chaitanya Arora
<b>Date &amp; time of test</b>	24/04/2022 23:15

<b>Result (failure)</b>	No output
<b>Possible causes</b>	We forgot to add a switch case for this scenario.
<b>Correction</b>	A switch case was added for this scenario, and the repeated test was successful.

## 10.3 Evaluation

<b>Name of the team member</b>	<b>Participation (%)</b>
Janibyek Bolatkhan	16.66%
Kasay Ito	16.66%
Chaitanya Arora	16.66%
Tushig Bat-Erdene	16.66%
Neda Radonjic	16.66%
Abdelrahman Desoki	16.66%

## 10.4 Protocol

Start (date & time)	Duration (hours)	Performer(s) name	Activity description
24/04/2022	1,5 hours	Chaitanya and Neda	Finishing the documentation and testing the code.
23/04/2022	2 hours	Janibyek	Worked on the prototype program by extending the skeleton code. Fixed bugs in the methods.
24/04/2022	1,5 hours	Desoki	Debug some methods and test the program.
24/04/2022	1 hour	Tushig	Reviewing the code and commenting methods and classes.
20/04/2022	1,5 hours	All team	Discuss the last review and check the methods that don't work, separate tasks, and schedule meetings.
23/04/2022	1 hour	Kasay	Fixing a bug