Battleship Bots

Problem Solving I

Q) What is the problem?

Q) What is the problem?

A) My ship keeps getting sunk…

Q) What is the problem?

A)   My ship keeps getting sunk…

Can we be a bit more precise?

What is that you would like your ship to do?

Maybe behave more like **Marks24**…

What does **Marks24** do?

Maybe behave more like **Marks24**…

What does **Marks24** do?

Watch **Marks24** for 5 minutes and then write down what it is doing on the Whiteboard…

So, what is **Marks24** doing?

So, what is **Marks24** doing?

Marks24

- Moves towards the nearest ship
- Fires at the nearest ship

So we set this as our goal

- Move towards the nearest enemy ship
- Fire at the nearest enemy ship

So we set this as our goal

- Move towards the nearest enemy ship
- Fire at the nearest enemy ship

In Software Engineering we refer to these as our requirements and we place them in a numbered list

Requirements

Number	Description	Priority
1	Move towards the nearest enemy ship	Medium
2	Fire at the nearest enemy ship	High

Note that we have added a **priority** to the requirements. It is more important that we fire at the nearest enemy that move towards them as hitting the nearest enemy gains you marks

Once we have our requirements how are we going to know if we have implemented them correctly?

Once we have our requirements how are we going to know if we have implemented them correctly?

We need to consider how we are going to test the code that we produce...

Each requirement requires at least one test associated with it to prove that the requirement has been implemented correctly

Each requirement requires at least one test associated with it to prove that the requirement has been implemented correctly

We call these tests the **Acceptance Tests**

Test 1	Move towards the nearest enemy ship
Pre-requisite	Battleship Bots server running
Requirements covered	1

Step	Action	Outcome
1	Start your Battleship	Your Battleship should move towards its nearest enemy

Test 2	Fire at the nearest enemy ship
Pre-requisite	• You have recorded the mark that your Battleship attained before this requirement was implemented • Battleship Bots server is re-started
Requirements covered	2

Step	Action	Outcome
1	Start your Battleship	
2	Wait 10 minutes	The mark attained by your Battleship is greater than the mark recorded before this requirement was implemented

In Software Engineering we refer to these tests as **Acceptance Tests**

They are always **agreed with the customer** at the beginning of a project

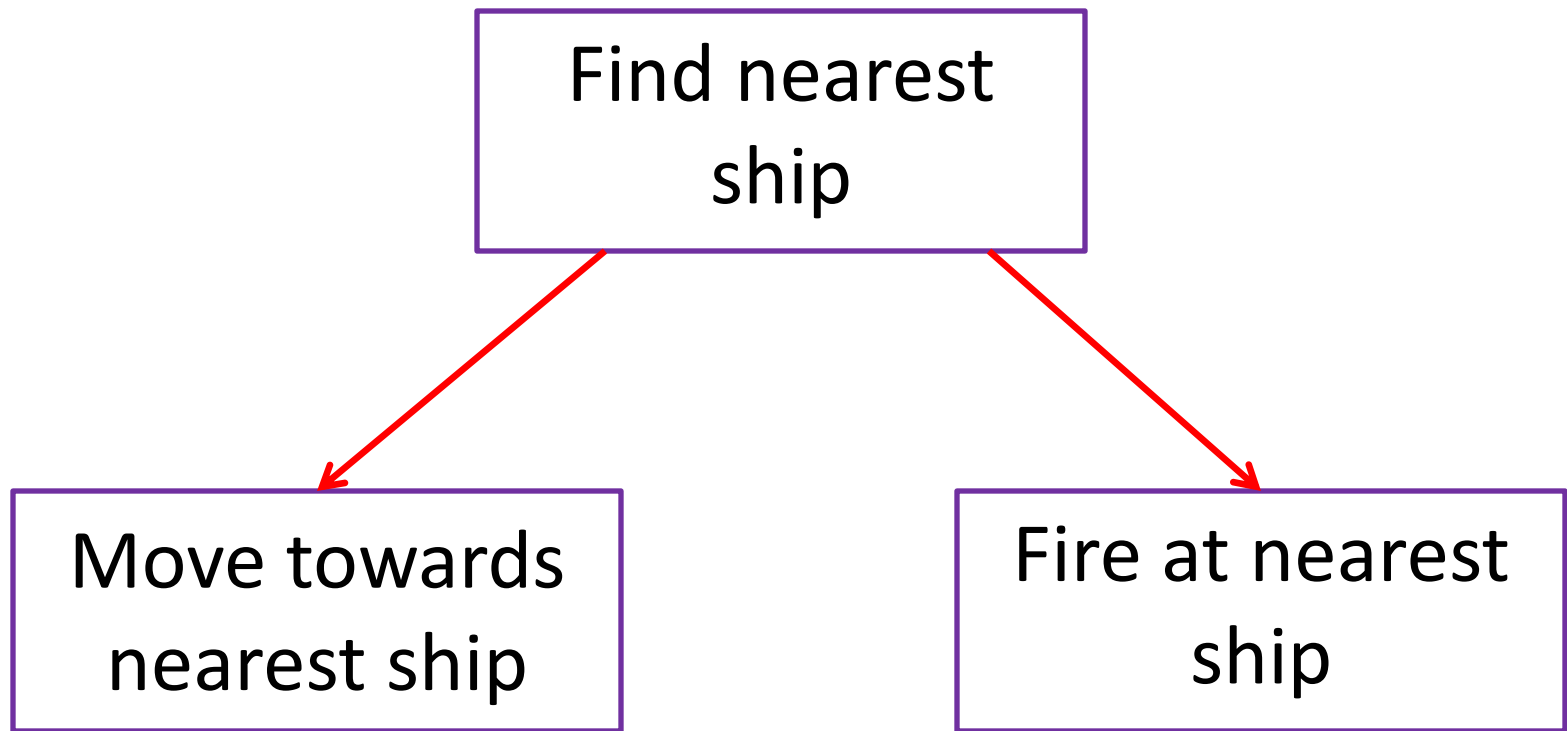They are used to **prove** that the required software has been delivered

They trigger **payment**

Next we need to think about how we are going to implement the requirements

Remember them…

1. Move towards the nearest enemy ship
2. Fire at the nearest enemy ship

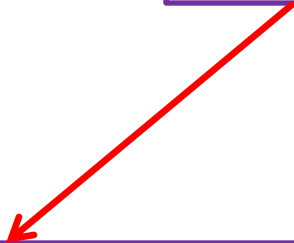In both requirements we need to know who the nearest enemy ship is...

Find nearest ship

Move towards nearest ship

Fire at nearest ship

But, how do you find the nearest?

You need to know the distance of each ship from yours!

```
┌─────────────────────┐
│   Calculate ship    │
│     distances       │
└─────────────────────┘
           │
           ▼
┌─────────────────────┐
│   Find nearest      │
│       ship          │
└─────────────────────┘
      ╱           ╲
     ▼             ▼
┌──────────────┐  ┌──────────────┐
│ Move towards │  │ Fire at nearest │
│ nearest ship │  │     ship      │
└──────────────┘  └──────────────┘
```
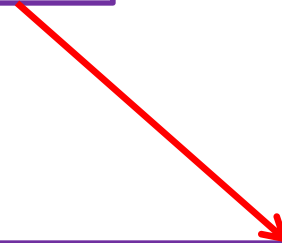
In Software Engineering we refer to this process as **Design**

Next we need to think about how to carry out **Implementation**

In tactics()…

You will need to call two routines

- calculate_ship_distances()
- find_nearest_enemy()

These routines will need to be written and placed above the tactics() routine

When these routines are written and placed in the correct place the **Acceptance Tests** can be ran

This process used to solve the Battleship problem can be viewed as follows...
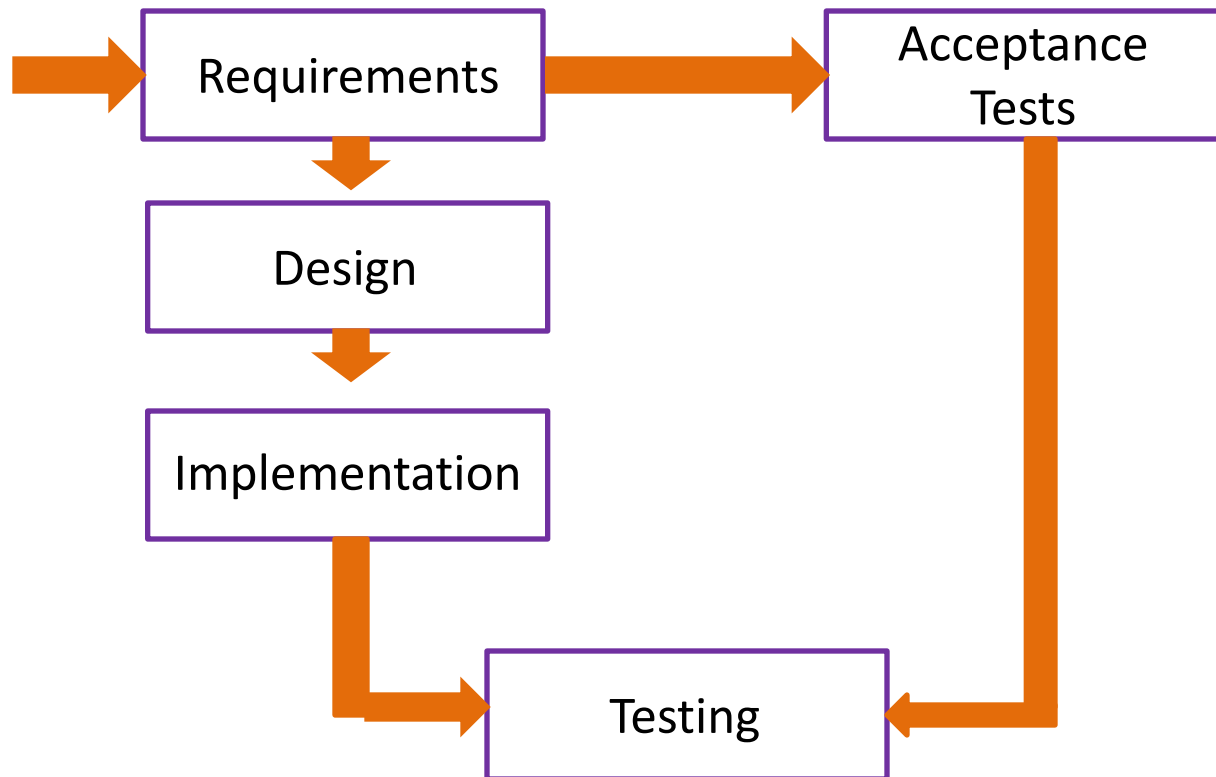
If you need help calculating ship distances or finding the nearest ship the you can…

- Ask your **tutor** for help
- Ask your **PAL leaders** for help
- Ask **EspressoProgramming** for help

The End