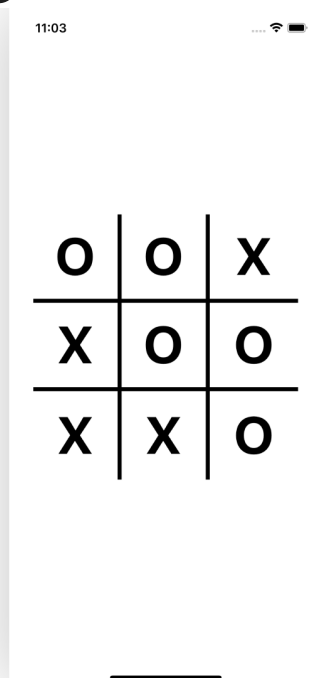# Responsive noughts and crosses UI

In this assignment, we will make the UI to a noughts and crosses game (you might know the game as "tic tac toe"). Now the temptation is to make this in a fixed size (e.g. each box is 100px wide and tall). But with this assessment, we want to create a responsive design that scales appropriately depending on the device resolution. It has to scale appropriately for this to be a pass.

Remember that you can view your app in multiple simulators, this will let you check your responsive design before you submit it.

How do I submit this?

This graded assessment is one of five graded assessments which will make up your midterm assignment submitted in week 10. I strongly advise you to complete it now as it is crucial to expanding your understanding of the subsequent weeks. Once you have completed it store it somewhere safe and do not share it with others. Instead, you can upload it with the other assignments during week 10.

Steps to complete:

This assessment doesn't have specific steps, but instead guidelines you should follow. Although the technical implementation is not as long, it might take a while to discover the correct approach.

- Your app UI should look identical to the screenshot but can vary in colour/style.
- You will need to use flex and flex-direction.
- You can take any approach to get the correct sized boxes, however, they must not be hardcoded values e.g. 100px and should be proportionate to the device width. They should also have a margin applied so that there is a gap between either side of the screen.
- Every element should be responsive, including the naughts and crosses which are just text elements inside views.
- For full marks make sure your styling is elegant and appropriate, make use of stylesheet styles, not inline.
- You should really combine all the skills we have learnt so far to make this app correctly.

- **You are not expected to program the logic of the game. You are just replicating the UI as seen in the screenshot.**

Mark scheme:

Description	Marks
UI	
No attempt made	0
The app has a basic UI that represents an attempt to replicate the screenshot	1
The app has a UI that somewhat represents the screenshot	2
The app has a UI that represents the screenshot well	3
The app has a UI that expands upon the screenshot, to make it more unique	4
The app significantly expands upon the screenshot	5
Responsiveness	
No attempt made	0
App uses fixed values, not responsive	1
App uses a mixture of fixed and responsive values	2
App uses fully responsive values (percentages)	3
App uses fully responsive values led by flex	4
App is fully responsive	5