

## **Project: Mobile App**

**Individual Assessment worth 35%**

**Title: Cumulative modification of code**

**Related outcomes from the unit outline:**

1. Apply advanced knowledge to analyse and design mobile software
2. Program a simple mobile application.
3. Interpret metrics produced by a relevant framework to build an effective mobile software product.

**Due date:** 27/10/2017, 11:59pm, WST.

**Suggested Length:** None specified.

**Submission instructions:**

To be submitted electronically via Blackboard electronic submission procedure

**Referencing:**

Not required

**Background:**

Assignment 1 gave you the opportunity to provide some base code for an app. The app is not particularly “smart” in how it plays the game. Extend your app by completing the following tasks:

**Task list:**

- Implement a winning strategy for the computer (figure 1).
- Keep track of the number of wins/losses/draws across separate invocations of the app i.e. persistent storage
- Allow these scores to be cleared
- Allow a human to undo his/her last move within five seconds of making a move
- Replay the last game in order of the moves made by each party
- Implement a three-level difficulty scheme where easy = computer plays randomly, hard = computer plays with logic in figure 1 and medium = computer uses random placement on odd turns and figure 1 on even turns
- Provide a document containing a record of the black box and unit tests you have completed to verify that your app works.

*If you or your opponent has two in a row, play on the remaining square.*

*Otherwise, if there's a move that creates two lines of two in a row, play that.*

*Otherwise, if the center square is free, play there.*

*Otherwise, if your opponent has played in a corner, play in the opposite corner.*

*Otherwise, if there's an empty corner, play there.*

*Otherwise, play on any empty square.*

Figure 1: Logic for winning strategy.