



# SuperGalgje

FHICT

<b>Niveau</b>	4 of 5.
<b>Leerdoelen</b>	Class, Property, Constructor, private/public, UI separation, algoritme.
<b>Vereiste voorkennis</b>	Method, GUI, Basic Types, If.
<b>Challenge Type</b>	Programming, algoritme.

## 1.1 Super-galgje

Galgje is een spel waarbij een speler het woord moet raden dan de computer in gedachten heeft. Opdracht: schrijf galgje en maak gebruik van de object georiënteerde mogelijkheden van C#.

1. Te programmeren classes: *Woord* en *Form1* (form).
2. Te programmeren property in de class *Woord*: *AantalLetters*.
3. Te programmeren methode in de classe *Woord*: `bool IsGoed(string woord)`
4. Te programmeren classes: *Woord*, *SpelStatus*, *Form1*.
5. *Form1* heeft een referentie naar 1 *SpelStatus*-object en geen referentie naar *Woord*.
6. Te programmeren properties in de class *Woord*: *AantalLetters* (read-only property).
7. Te programmeren methoden in de class *Woord*: `bool IsGoed(string woord)`.

8. Te programmeren property in de class *SpelStatus*: *HetWoord* van het type *Woord* (dus NIET van het type `string`).
9. Verder kun je in de class *SpelStatus* methodes en/of properties toevoegen die de status van het spel zoals het aantal geraden letters bijhouden.

### Eisen voor gevorderden:

1. Programmeer er nog een class *Speler* bij en zorg ervoor dat je met twee personen het spel tegen elkaar (tegen de computer) kunt spelen.
2. Class *Speler* heeft een property van het type *SpelStatus* en diverse methodes die jij zelf bedenkt. Het form krijgt 2 *Speler*-objecten en verder geen enkel ander object.