## W6.2 StopWatch(C#) 1:00u

### Inleiding

#### Leerdoel

* Een DispatcherTimer kunnen instellen
* Begrijpen hoe een timer werkt
* Kennismaken met maken van Properties

#### Startsituatie

* Startprogramma staat gereed in C#

### Oefening (C#)

* Voeg de volgende fields toe in de MainWindow class:
  + \_timer (DispatcherTimer)
  + \_ronde (int)
  + \_numberOfEvents (int)
* Initialisatie
  + Zorg dat er bij het initialiseren van de class een DispatchterTimer wordt gemaakt en wordt bewaard in het field \_timer.
  + De DispatcherTimer moet zodanig worden ingesteld dat er elke milliseconde een *tick* event wordt gegenereerd. Zorg er ook voor dat je een Eventhandler maakt voor dit click-event.
* EventHandler Tick event
  + De eventhandler handelt het click event van de timer als volgt af:
    - Het aantal events (\_numberOfEvents) wordt met 1 verhoogd en in het MainWindow getoond,
    - Het aantal uren, minuten, seconden en milliseconden dat is verstreken sinds de DispatcherTimer is gestart, wordt in het MainWindow bijgewerkt.

Tip: declareer **properties** voor het aantal uren, minuten, seconden en milliseconden, geef deze bij het starten van de DispatcherTimer allemaal de waarde 0, en verhoog deze bij het verwerken van het click event (zorg wanneer milliseconden > 999 is, de seconden met 1 worden verhoogd en de milliseconden op 0 worden gezet; etc.).

* Button Start
  + Wanneer de gebruiker op de Butten Start drukt en de timer is nog niet aan het lopen, worden de volgende handelingen uitgevoerd:
    - Het aantal ronden (\_ronden) en het aantal events (\_numberOfEvents) wordt op 0 ingesteld
    - Wordt de Listbox leeg gemaakt
    - De Button Ronde wordt enabled
    - De Button Start wordt disabled
    - De DispatcherTimer \_timer wordt gestart
* Button Ronde
  + Wanneer de gebruiker op de Button Ronde drukt en de timer aan het lopen is:
    - Wordt het aantal afgelegde ronden met 1 verhoogd
    - Wordt bovenaan in de ListBox een regel toegevoegd, met daarin de tekst welke ronde is afgelegd en wat de totaal verstreken tijd was
* Button Stop
  + Wanneer de gebruiker op de Button Stop drukt en de timer aan het lopen is:
    - Wordt de timer gestopt
    - Wordt bovenaan in de ListBox een regel toegevoegd, met daarin de tekst dat de laatste ronde is afgelegd en wat de totaal verstreken tijd was
    - Wordt de knop Start enabled en de knoppen ronde en stop disabled

### Verdieping

Zoals je waarschijnlijk is opgevallen, loopt de verstreken tijd die je in het MainWindow toont, achter bij de werkelijke tijd. De DispatcherTimer kan niet garanderen dat hij exact elke milliseconde (zoals ingesteld) een click event genereert.

Dit lossen we op door de systeem tijd bij het starten van de Dispatcher timer op te vragen, en vervolgens de verstreken tijd in het Window te laten zien. Voer de volgende aanpassingen door

* Als je dat gedaan hebt, kun je de variabelen voor het aantal uren, minuten, seconden en milliseconden verwijderen.
* Declareer een field voor de starttijd. Dit field is van het type DataTime.
* Pas de functionaliteit van de Start Button aan:
  + Zorg ervoor dat het field voor de tijd krijgt waarop de gebruiker op de Start knop drukte.

NB: Zoek in de documentatie bij DateTime op, welke eigenschap je hoervoor kunt gebruiken

* Pas de functionaliteit van de Ronde en Stop Button aan:
  + bereken het verschil tussen de start tijd en de huidige tijd. Het resultaat is een TimeSpan (die we ook gebruiken op de frequentie van de DispatcherTimer in te stellen). Gebruik dit verschil om de tekst voor de ListBox te vullen
  + Ga na dat dit verschil inderdaad een TimeSpan is.
* Pas de functionaliteit van het Tick event aan:
  + bereken het verschil tussen de start tijd en de huidige tijd. Het resultaat is een TimeSpan (die we ook gebruiken op de frequentie van de DispatcherTimer in te stellen).
  + Ga na dat dit verschil inderdaad een TimeSpan is.
  + Zoek in de documentatie op, hoe je uit deze timespan het aantal uren, minuten, seconden en milliseconden ophaalt.
  + Zet het aantal verstreken uren, minuten, seconden en milliseconden in het MainWindow