Lars van Valen 4158083

Wouter Sondagh 4143582

De door ons gedane wijzigingen:

Hoofdscherm.cs:

* libary toegevoegd (system.IO), nodig om in deze klasse te kunnen openen vanuit een bestand.
* Onze namen even toegevoegd in het ‘about’ venster.
* Openen methode toegevoegd. In eerste instantie waren het twee methodes, één voor afbeelding en één voor txt. Zorgde voor veel extra code en programma zelf werd rommeliger (je had namelijk een openen button en een openen speciaal button). Dit weten samen te voegen. Nu kan de gebruiker kiezen in het dialoog tussen .txt, .jpg, .bmp en .gif.
* Veel moeite gehad met het uitwerken van openen vanuit een .txt. Daarom is het helaas een lange lap code geworden. Maar het werkt nu wel! Voor overzicht paar regions neergezet.

Objecten.cs

* Deze klasse is nieuw. In deze klasse worden objecten gemaakt van een bepaalde typen. We hebben gekozen voor een abstracte klasse ‘MaakObject’ met daarin alle benodigde variabelen voor de teken objecten. Ook zit hier een abstracte methode in voor tekenen en geklikt. Deze moeten de subklassen implementeren. Ook zijn er 2 formules te vinden die nodig zijn om te kijken of je op een cirkel klikte en wat de afstand was tot een lijn. Alle klasses in Objecten.cs erven van MaakObject
* Alle ervende klassen krijgen parameters mee die de nodige info bevatten om een object te maken van dat type. Deze parameters worden opgeslagen in de bijbehorende variabelen in de MaakObject klasse.
* Bij Tekstobject klasse hebben we gekozen om een bij het klikken uit te gaan van een rechthoek om de letter heen. Voordeel hiervan is dat de letter makkelijk weg te halen is. Nadeel is dat de rechthoek uit gaat van ‘lange’ letters zoals je ‘j’. Je kan dus onder de meeste letters klikken omdat de rechthoek wat groter is. Bewust zo gelaten omdat je anders krijgt dat je bij de ‘lange’ letters niet op het staartje kan klikken. Dit vonden we zwaarder wegen dan een stukje onder een niet ‘lange’ letter klikken.
* Rest van de klassen spreken aardig voor zich denken we. Bij aantal klassen hebben we nog een correctie toegevoegd zodat je niet precies op het lijntje hoeft te klikken maar er een klein stukje vanaf mag zitten.

Program.cs

* Geen aanpassingen

Schets.cs

* Lijst waarin de objecten komen toegevoegd
* Setbitmap en getbitmap properties toegevoegd. Hierdoor kunnen we vanuit andere klasses de bitmap aanpassen/opvragen.
* Methode zet in lijst. Krijgt in parameter een MaakObject object mee, en zet deze in de lijst. Daarna roept hij TekenLijst aan.
* TekenLijst gaat door de hele lijst hen en tekent elk object dat hij tegen komt.
* 2 tal properties om respectievelijk de grootte van de lijst op te vragen en objecten uit de lijst te getten en te setten.
* RemoveLastFromList checkt eerst of er wel elementen in de lijst zitten en verwijderd daarna de laatste. Hiermee werkt de undo knop.

SchetsControl.cs

* GetBitmap en SetBitmap properties. Deze werken via de properties die we al bij schets hebben toegevoegd. Door hem ook bij schetscontrol toe te voegen konden we er op veel meer plaatsen bij zonder aan schets te komen.
* Schoon methode aangepast. Verwijderd nu ook alle objecten uit de lijst.
* Undo methode: maakt gebruik van de RemoveLastFromList en verwijderd zo de laatste uit de lijst. Tekent daarna alles in de lijst opnieuw.
* Methode ZetInLijst om objecten toe te voegen aan de lijst in schets.
* Property lijstgrootte omdat niet elke klasse bij schets kan maar de dan vaak wel bij schetscontrol. Property maakt gebruik van property in schets.
* TekenLijst zelfde verhaal als bij Lijstgrootte, nu alleen een methode die Tekenlijst in schets aanroept.
* Zetschetsobject, bij het openen van een tekst file wordt een schets aangemaakt, die moet de zelfde schets worden als de schets die tekend. Dat wordt in deze property mogelijk gemaakt.

SchetsWin.cs

* System.IO toegevoegd nodig voor opslaan.
* In constructor aangegeven dat er een undo button moet komen, deze wordt gemaakt in MaakUndobutton(). Bij klikken op de knop wordt dit afgehandeld door de methode in Schetscontrol.
* Opslaan button toegevoegd in MaakfileMenu()
* Opslaan methode. Zelfde verhaal als bij openen, dit waren eerst twee methodes, later samengevoegd zodat er minder code is en je een overzichtelijker programma krijgt.
* Setbitmap property nodig om vanuit hoofdscherm te kunnen openen
* Zetschetsobject property nodig om vanuit hoofdscherm de zelfde schets te krijgen als waarmee het venster werkt.

Tools.cs

* Bij elke tool wordt nu ipv getekend de tool in de lijst gezet. Dit zie je dus bij elke tool terug.
* Compleet methode eruit gesloopt. Was niet meer nodig
* Volrechthoek erft niet meer van rechthoek maar nu meteen van tweepunttool. Dit omdat er anders 2 objecten in de lijst werden geplaatst. Namelijk die van volrechthoek en die uit de base (rechthoek). Dit is nu opgelost door andere ervingsstructuur.
* Bij elke tool MuisLos toegevoegd, in deze methode wordt de tool aan de lijst toegevoegd.
* Cirkeltool en Volcirkeltool toegevoegd