

ROC van Twente, Applicatieontwikkeling

Gieterij 200

7553VZ Hengelo  
2016-2017

Project Bankieren

# Voorwoord

Tijdens het vierde semester van jouw opleiding staan er voor het vak programmeren twee projecten op het programma, waarvan dit het eerste project is. In je studiehandleiding kun je lezen hoe dit project wordt gewaardeerd.

Tijdens dit project ga je een WPF-applicatie voor bankieren realiseren. De projectgroep wordt samengesteld uit twee studenten.

Het project biedt de mogelijkheid om de applicatie te realiseren in drie moeilijkheidsniveaus. Je mag zelf je niveau bepalen. Het eerste niveau is eenvoudig, je kunt van je eigen betaalrekening geld overschrijven naar je spaarrekening en weer terug. Het tweede niveau vraagt meer inspanningen, de applicatie beschikt over een bank met cliënten. Naast het overschrijven van geld naar je eigen rekening is het ook mogelijk geld over te schrijven naar andere cliënten van de bank. Het derde niveau is het hoogste niveau, naast de hierboven genoemde functionaliteiten moeten er een aantal aanvullende functionaliteiten voor de bank worden geschreven.

**Waardering.**

Omdat de moeilijkheidsgraad per niveau verschilt, zullen de volgende waarderingen plaats vinden:

* Niveau 1: voldoende = maximaal 6
* Niveau 2: goed = maximaal 8
* Niveau 3: uitmuntend = maximaal 10

**Het ontwerp**

Omdat je tijdens het vierde semester van je opleiding vooralsnog onvoldoende ervaring hebt kunnen opdoen met het ontwerpen van een dergelijke applicatie, hebben we een klassendiagram voor je ontworpen. In bijlage 1 is het klassendiagram weergegeven. Wanneer je niveau 3 wilt gaan uitwerken, is het toegestaan om methoden aan het klassendiagram toe te voegen. Doe dit echter wel in overleg met de projectleider (docent).

**Duur en het verloop van het project**

Het project start maandag 6 maart en duurt tot vrijdag 24 maart. Naast de contacturen is het toegestaan om thuis of op school verder te werken aan het project. Tijdens het project moet een logboek worden bijgehouden en een planning worden gemaakt. Het project dient uiterlijk vrijdag 24 maart voor 17:00 uur te worden ingeleverd in natschool in de map ‘Inleveropdracht Project Bankieren’.

**Oplevering van het project**

Het project wordt opgeleverd door middel van een presentatie tijdens de toetsweek. Beide projectleden presenteren ieder hun deel van de applicatie. Na afloop van de presentatie kunnen er vragen worden gesteld door de docent.

Inhoudsopgave

[Voorwoord 1](#_Toc476152524)

[Inleiding 3](#_Toc476152525)

[Functionaliteiten 4](#_Toc476152526)

[Het eerste niveau. 4](#_Toc476152527)

[Het tweede niveau. 5](#_Toc476152528)

[Het derde niveau. 6](#_Toc476152529)

[Opdrachten 7](#_Toc476152530)

[Aandachtspunten voor de presentatie. 7](#_Toc476152531)

[Bijlage 1 - Klassendiagram. 9](#_Toc476152532)

# Inleiding

Met de applicatie Bankieren kun je al je bankzaken verrichten. Zoals je in het voorwoord hebt kunnen lezen word je de mogelijkheid geboden om de applicatie te realiseren op een niveau naar keuze. Hieronder kun je de functionaliteiten van de applicatie per niveau bestuderen. In bijlage 1 is het klassendiagram weergegeven.

Met de opdrachtgever is afgesproken dat de applicatie wordt gerealiseerd in WPF en geschreven in C#.

**Algemene informatie over de applicatie.**

De applicatie heeft een tweelagen structuur. De eerste laag, Front- layer genaamd, is de laag waarmee je communiceert met de cliënt van de bank. De tweede laag is de Business- layer, hierin bewaar je al je klassen. Wanneer we ons in de toekomst gaan bezighouden met databases, gaan we ons verdiepen in de three layer architecture. Een voorbeeld van een programma met een tweelagen structuur is meegeleverd met het project. Bestudeer de werking van het programma alvorens je je gaat bezighouden met de applicatie Bankieren. In de voorbeeld applicatie zitten technieken die je goed van pas kunnen komen tijdens het project.

De applicatie Bankieren beschikt **niet** over een database. Studenten die de opdracht gaan realiseren op niveau 2 of 3 moeten echter wel beschikken over een datacollectie in de vorm van cliënten van de bank (List<Bankrekeninghouder>). Een oplossing hiervoor is een statische verzameling van bankrekeninghouders die beschikbaar komt wanneer men de applicatie opstart. Dit is echter wel statisch, telkens wanneer de applicatie opnieuw wordt opgestart wordt weer dezelfde collectie aangeboden. Zie het meegeleverde voorbeeld programma.

De niveaus zijn stapelbaar, wanneer je ervoor kiest niveau 3 uit te werken moet je ook niveau 1 en 2 uitwerken.

# Functionaliteiten

## Het eerste niveau.

Voor het eerste niveau moet de applicatie beschikken over de volgende functionaliteiten.

1. **Inloggen:** De cliënt logt in op de applicatie en wordt doorverwezen naar de hoofd- window. Hij ziet hier zijn persoonsgegevens zoals voor- en achternaam en burger- servicenummer (bsn). Het burgerservicenummer moet voldoen aan de 11- proef. De cliënt kan op de hoofd- window navigeren naar de spaar- en betaalrekening- window. Studenten die niveau 1 gaan realiseren loggen op een andere manier in dan studenten die niveau 2 of 3 realiseren. Voor niveau 1 neem je een class Inloggen op in de Business- layer (niet in het klassendiagram opgenomen). De class Inloggen beschikt over twee methoden. De methode “Cliënt()” retourneert een instantie van de class Bankrekeninghouder. Deze instantie omvat je bankrekeningen waarmee je gaat bankieren. Ook bevat deze instantie persoonsgegevens waaronder gebruikersnaam en wachtwoord. Schrijf een methode “Login()” met de volgende methode definitie “Login(gebruikersnaam : string, wachtwoord : string) : bool” om in te loggen. De methode controleert of de opgegeven gebruikersnaam en wachtwoord overeenkomen met die van de instantie.
2. **Betaalrekening:** Voor de betaalrekening is een speciaal window opgenomen in de applicatie. Met de betaalrekening kun je geld overboeken naar je spaarrekening. Na het overboeken wordt het saldo van de bankrekening direct aangepast. Houd rekening met het maximale krediet van de betaalrekening. Wanneer het maximale krediet wordt overschreden, moet hiervoor een waarschuwing komen, en zal de transactie niet worden uitgevoerd. Na het sluiten van de betaalrekening- window wordt de cliënt weer naar de opstart- window genavigeerd.
3. **Spaarrekening:** Voor de spaarrekening is er een speciaal window opgenomen. Met de spaarrekening kun je geld overboeken naar de betaalrekening. Na het overboeken wordt het huidige spaarsaldo direct bijgewerkt. Houd er rekening mee dat je niet meer geld kunt overboeken dan je op de spaarrekening hebt staan. Tevens ontvang je rente op je spaarrekening. De spaarrekening beschikt over een functionaliteit die de opgebouwde rente tot dus ver (van het huidige jaar) kan tonen (niet opgenomen in het klassendiagram). Deze kun je echter niet zelf bijschrijven, dit doet de bank aan het einde van het jaar. Na het sluiten van de spaarrekening- window wordt je weer naar de hoofd- window genavigeerd.
4. **Applicatie sluiten**: De applicatie kan worden gesloten door de hoofd- window te sluiten.

## Het tweede niveau.

Voor het tweede niveau moet de applicatie beschikken over de volgende functionaliteiten.

1. **Hoofd- window:** Na het inloggen wordt de cliënt door verwezen naar de hoofd- window. De cliënt ziet hier zijn persoonsgegevens zoals voor- en achternaam en burgerservicenummer (bsn). Het burgerservicenummer moet voldoen aan de 11- proef. Tevens kan de cliënt navigeren naar de spaar- en betaalrekening- window.
2. **Betaalrekening:** Met de betaalrekening kun je geld overboeken naar je spaarrekening of naar een betaalrekening van een andere bankcliënt. Na het overboeken wordt het saldo van de bankrekening afgeschreven en wordt het huidige saldo direct bijgewerkt en getoond. Houd rekening met het maximale krediet. Wanneer het maximale krediet wordt overschreden, moet hiervoor een waarschuwing komen, en zal de transactie niet worden uitgevoerd. Voor het overboeken van een betaling naar een andere cliënt kan men een cliënt selecteren uit een ComboBox om zodoende het rekeningnummer van de cliënt te achterhalen. Hiervoor mag men een class met methoden toevoegen aan de business- layer. Na het sluiten van de betaalrekening- window word je weer naar de hoofd- window genavigeerd.
3. **Spaarrekening:** Voor de spaarrekening is er een speciaal window opgenomen. Met de spaarrekening kun je geld overboeken naar de betaalrekening. Na het overboeken wordt het huidige spaarsaldo direct bijgewerkt. Houd er rekening mee dat je niet meer geld kunt overboeken dan je op de spaarrekening hebt staan. Tevens ontvang je rente op je spaarrekening. De spaarrekening beschikt over een functionaliteit die de opgebouwde rente tot dus ver (van het huidige jaar) kan tonen. Deze kun je echter niet zelf bijschrijven, dit doet de bank aan het einde van het jaar. Na het sluiten van de spaarrekening- window wordt de cliënt weer naar de hoofd- window genavigeerd.
4. **Applicatie sluiten**: De applicatie kan worden gesloten door de hoofd- window te sluiten.

# Het derde niveau.

Voor het derde niveau moet de applicatie beschikken over alle functionaliteiten van niveau 2 aangevuld met de functionaliteiten zoals hieronder genoemd. De applicatie beschikt nu ook over een bank welke haar cliënten kan bedienen.

1. **Inloggen:** De applicatie is nu toegankelijk voor de bankbedienden en cliënten van de bank. Om in te loggen moet men de juiste rol aanvinken middels RadioButtons. De bank heeft een vaste gebruikersnaam en wachtwoord voor alle bankbedienden.
2. **Toevoegen van cliënt:** De bankbediende heeft een mogelijkheid om nieuwe cliënten toe te voegen aan het cliëntenbestand (List<Bankrekeninghouder>).
3. **Verwijderen van een cliënt:** De bankbediende heeft de mogelijkheid om een cliënt te verwijderen uit het cliëntenbestand (List<Bankrekeninghouder>).
4. **Rente bijschrijven:** De bankbediende heeft de mogelijkheid om de jaarlijkse rente voor alle cliënten bij te schrijven. Dit gebeurt niet automatisch maar is een actie die de bankbediende zelf moet ondernemen.

# Opdrachten

1. Maak een keuze uit een van de drie niveaus en bestudeer vervolgens het klassendiagram. Ontwerp het use case diagram dat hoort bij jouw keuze.
2. Zoek een geschikt MockUp programma en maak van alle windows uit je applicatie een Mockup. Bespreek je MockUp’s met je docent. Laat middels een tekening zien hoe je door de windows van je applicatie kunt navigeren.
3. Realiseer de applicatie samen met je projectpartner.
4. Voorzie de code van gedegen commentaar.
5. Schrijf een Unit- test voor de methoden van de klassen uit de Business- layer.
6. Bereid je presentie voor.

# Aandachtspunten voor de presentatie.

Tijdens de presentatie moet je aantonen dat je applicatie naar behoren werkt. Daarvoor moet je de applicatie in een chronologische volgorde doorlopen. Hieronder per niveau van uitvoering aan aantal tips.

**Niveau 1**

1. Toon aan dat je kunt inloggen.
2. Laat zien dat je persoonsgegevens zoals voor- en achternaam en bsn nummer worden getoond op de hoofd- window.
3. Laat in de code zien dat je bsn- nummer voldoet aan de 11 proef.
4. Laat zien dat je vanaf je hoofd- window kunt navigeren naar je betaalrekening. Maak vervolgens meer geld over naar je spaarrekening dan je maximale krediet. Laat zien hoe deze fout wordt afgehandeld. Maak hierna een bedrag, kleiner dan je maximale krediet, over naar je spaarrekening. Toon aan dat het huidige saldo van je betaalrekening is bijgewerkt.
5. Laat zien dat je vanaf je hoofd- window kunt navigeren naar je spaarrekening. Maak vervolgens meer geld over naar je betaalrekening dan je op de spaarrekening hebt staan. Laat zien hoe dit wordt afgehandeld. Maak vervolgens een kleiner bedrag over naar je spaarrekening. Toon aan dat het geld van je spaarrekening is afgeschreven en is bijgeboekt op je betaalrekening.
6. Sluit je programma af op de hoofd- window.

**Niveau 2**

1. Toon aan dat je minimaal kunt inloggen op twee verschillende bankrekeningen.
2. Laat zien dat je persoonsgegevens zoals voor- en achternaam en bsn nummer worden getoond op de hoofd- window.
3. Laat in de code zien dat je bsn- nummer voldoet aan de 11 proef.
4. Toon aan dat je net als bij niveau 1, onder punt 4 en 5 geld kunt overmaken naar je eigen bankrekeningen.
5. Toon aan dat je geld kunt overmaken naar verschillende cliënten van de bank. Laat zien dat het geld van jouw rekening wordt afgeschreven en wordt bijgeschreven op de rekening van de begunstigde.
6. Sluit je programma af op de hoofd- window.

**Niveau 3**

1. Toon alle punten van niveau 2 aan.
2. Toon aan dat je kunt inloggen als bankbediende.
3. Voeg als bankbediende een cliënt toe aan het bestand van cliënten. Toon aan dat de 11 proef werkt door een ongeldig bsn nummer in te voeren. Navigeer terug naar de inlog- window en log in als de nieuwe cliënt. Let op! Sluit de applicatie niet af, anders gaan je gegevens verloren.
4. Log weer in als bankbediende en verwijder de zonet toegevoegde cliënt. Toon aan dat je niet meer kunt inloggen met de gegevens van de verwijderde cliënt.
5. Toon aan dat je rente kunt beschrijven aan alle cliënten van de bank.
6. Sluit je programma af op de hoofd- window.

# Bijlage 1 - Klassendiagram.

