



# basic education

---

Department:  
Basic Education  
**REPUBLIC OF SOUTH AFRICA**

## **INLIGTINGSTEKNOLOGIE**

### **RIGLYNE VIR PRAKTIESE ASSESSERINGSTAKE**

**2012**

**Hierdie riglyne bestaan uit 28 bladsye.**

# **Inligtingstechnologie**

## **Praktiese Assesseringstaak (PAT)**

**2012**

### **Riglyne aan die onderwyser**

**Hierdie afdeling bestaan uit 7 bladsye.**

## A Inleiding

Die doel van die Praktiese Assesseringstaak (PAT) is om die onderwyser die geleentheid te gee om die toegepaste bevoegdheid van die leerder direk en sistematies waar te neem en te evalueer. Die PAT behels die toepassing van kennis en vaardighede.

In Inligtingstegnologie tel die PAT 25% van die totale promosiepunt vir die vak (d.i. 100 uit die 400 punte). Dit word oor die eerste drie kwartale van die graad 12-skooljaar geïmplementeer en moet as een uitgebreide taak onderneem word wat in verskillende fases of 'n reeks kleiner aktiwiteite opgedeel word.

Die IT-PAT bestaan uit drie fases:

- Onderzoek en analise
- Ontwerp
- Kodering, implementering en interne dokumentasie, insluitend aanbieding en algemene evaluering

## B Programmeringsprojek

### 1. Waaroor gaan die programmeringsprojek?

Die programmeringsprojek van die PAT in Inligtingstegnologie vereis dat die leerders die ontwikkelingsinstrumente wat hulle bestudeer het, moet gebruik om 'n programmatuur-oplossing vir 'n spesifieke probleem binne 'n gegewe scenario te ontwikkel.

### 2. Beplanning van die programmeringsprojek

#### 2.1 Beskrywing

##### Scenario vir die 2012-graad 12-PAT:

Daar word van die leerder verwag om 'n program te skryf wat as 'n inligtingstelsel deur die personeel en/of besoekers van enige natuurgeoriënteerde toeriste-aantreklikheid in Suid-Afrika gebruik kan word, byvoorbeeld:

- 'n Natuurresewaat
- 'n Wildresewaat
- 'n Akwarium
- 'n Voëlpark
- 'n Dieretuin
- Enige ander wildlewe- of natuurgeoriënteerde toeriste-aantreklikheid in Suid-Afrika

Die fokus van die stelsel moet wees om natuurverwante inligting aan personeel en/of besoekers te verskaf.

**LET WEL:** Dit moet **NIE** 'n besprekingstelsel wees **NIE**.

Die leerder moet navorsing doen oor die inligtingsbehoefte van die personeel en/of besoekers van die toeriste-aantreklikheid van sy/haar keuse. 'n Funksionele en bruikbare stelsel moet ontwerp word wat suksesvol by die spesifieke toeriste-aantreklikheid geïmplementeer kan word.

Die leerder mag sy/haar eie onderwerp/toepassing binne die konteks van die gegewe scenario kies.

Met die voltooiing van die projek sal die leerder die volgende vaardighede toepas:

- Onderzoek
- Analise
- Ontwerp
- Ontwikkeling van programmatuur
  - Programmeringsvaardighede deur die gebruik van die programmeringstaal wat bestudeer is
  - Ontwikkeling van 'n databasis
  - Ontwerp van 'n grafiese gebruikerskoppelvlak (GUI)

Gedeeltes van die assesseringsinstrument sal projekte penaliseer wat nie aan die gegewe scenario verwant is nie.

## 2.2 Ontwikkelingsfases van die projek

Die programmeringsprojek moet in drie fases voltooi word, soos aangedui in die volgende tabel:

FASES	PUNTE	%
Fase 1: Onderzoek en analise	30	17,5%
Fase 2: Ontwerp	30	17,5%
Fase 3: Kodering, implementering en interne dokumentasie, insluitend aanbieding en evaluering	110	65%
<b>TOTAAL</b>	<b>170</b>	<b>100</b>

Dokumentasie/Bewys van wat die leerder gedurende elke fase gedoen het, moet volgens gespesifiseerde intervalle ingelewer word. Dit is uiters belangrik dat bewyse van die werk in al die fases voorsien word. Die dokumentasie wat vir elk van hierdie fases vereis word, word in die leerderafdeling van hierdie dokument verskaf.

**Sperdatums** vir die inlewering van die finale produk van elke fase sal deur die onderwyser vasgestel word. Die produkte van elke fase sal geassesseer word en die punte sal aangeteken word.

## 2.3 Vereistes van die projek

Die leerder moet aandag gee aan die volgende minimum kriteria:

### Fase 1: Onderzoek en analise

- Beskrywing van die probleem in sy/haar eie woorde waarin die hoofaspekte in een paragraaf uitgelig word
- Onderzoek van die onderwerp deur:
  - Identifisering van die aard en die omvang van die probleem en insameling van feite
  - Verkryging van inligting vanuit 'n werklike situasie waarin die programmatuurprodukt deur die eindgebruiker gebruik gaan word
- Analisering van die probleem: Wat is die vereistes en wat moet deur die programmeringsoplossing voorsien word?

(Sien Leerderafdeling 2 Fase 1 en Assesseringsinstrument Fase 1)

## Fase 2: Ontwerp

- Ontwerp 'n oplossing: Hoe sal die program/stelsel aan die vereistes voldoen? Voorsien 'n goed beplande oplossing vir die probleem.  
(Sien Leerderafdeling 2 Fase 2 en Assesseringsinstrument Fase 2)

## Fase 3: Kodering, implementering en interne dokumentasie

- Die projek moet die belangrikste ontwikkelingsinstrumente insluit, m.a.w. 'n databasis en programmeringstaal wat toepaslik geïntegreer is. (Ander toepassings kan saam met hierdie ontwikkelingsinstrumente geïntegreer word.)
- Ander aspekte van die programmeringsprojek wat geassesseer sal word, sluit in:
  - Programmeringstyl
  - Grafiese gebruikerskoppelvlak (GUI)
  - Gebruik van mens-rekenaar-interaksiebeginsels ('HCI') en programmatuur-ingenieursbeginsels
  - Funksionaliteit van die program
  - Vlak van kundige programmering
  - Robuustheid van die program, insluitend die gebruik van defensiewe programmeringstegnieke
  - Of die projek ooreenstem met die oorspronklike mikpunte en doelwitte
- Interne dokumentering om dele van die program te verduidelik

## Aanbieding en algemene evaluering

- Evalueer die volgende:
  - Tydbestuur van die leerder – Het hy/sy al die sperdatums nagekom?
  - Die leerder se vermoë om dele van kode wat ewekansig gekies word, te verduidelik
  - Geskiktheid van die oplossing binne die konteks van die scenario(Sien Leerderafdeling 2 Fase 3 en Assesseringsinstrument Fase 3)

## 3. Instruksies aan die leerders

Verwys na Leerderafdeling van hierdie dokument.

## 4. Hulpbronne

Die leerder sal die volgende hulpbronne nodig hê om die projek te voltooi:

- Toegang tot 'n rekenaar met die volgende programme:
  - Programmeringstaal: Java of Delphi
  - Kantoortoepassingspakket, soos 'n woordverwerker (bv. MS Word) en 'n aanbiedingspakket (bv. MS Powerpoint)
  - Databasisprogrammatuur, soos MS Access of MySQL
- IDE (vir Delphi is dit deel van die programmeringstaal, maar vir Java het jy addisionele programmatuur nodig, soos Netbeans)

## 5. Assesering van die PAT

Die projek moet onder gekontroleerde toestande voltooi word en deur die onderwyser gefasiliteer en deurlopend gemonitor word.

Sien Assesseringsinstrumentafdeling vir die assesseringsblaaie vir die verskillende fases.

Onderwysers moet seker maak dat leerders die volgende dokumente aan die begin van hulle graad 12-jaar ontvang:

- Die 'Instruksies aan die leerders'-afdeling wat in hierdie dokument ingesluit is
- Die assesseringsblaaie vir al die fases wat in hierdie dokument ingesluit is

Leerders moet toegelaat word om nabetrugting te hou oor die punte wat hulle gekry het en om aandag te gee aan die foute wat hulle gemaak het, voordat hulle met die volgende fase voortgaan. Die punte wat aanvanklik vir die evaluering van 'n spesifieke fase toegeken is, sal NIE na die nabetrugting hersien word NIE.

Daar sal van leerders verwag word om hulle stelsel aan die einde van Fase 3 te demonstreer. Onderwysers moet die projekte volgens die assesseringsinstrument wat vir Fase 3 voorsien is, assesseer.

Leerders sal NIE toegelaat word om die onderwerpe van hulle projekte te verander nadat Fase 1 en 2 voltooi en geassesseer is NIE. Indien 'n leerder wel besluit om sy/haar onderwerp te verander nadat Fase 1 of 2 voltooi is, moet die leerder die vroeëre fases vir die nuwe onderwerp oordoen. In hierdie geval sal die onderwyser die opgedateerde Fase 1 en 2 NIE weer assesseer NIE. Die punte vir die oorspronklike fases moet aangeteken word. Fase 3 sal egter NIE geassesseer word tensy Fase 1 en 2 navorsing weerspieël wat vir die nuwe onderwerp gedoen is nie.

Die verband tussen al die fases van 'n projek moet streng en deurlopend tydens assessering gekontroleer word. Bewyse van werk wat in vorige fases gedoen is, moet altyd gedurende assessering en moderering van 'n spesifieke fase van ontwikkeling beskikbaar wees.

### **Riglyne vir die demonstrasie en interne evaluering van die projek:**

- Die onderwyser moet datums en tye vir demonstrasies skeduleer. Laat ongeveer 15 minute per projek toe vir die demonstrasie, asook 5 minute vir die opstelling van die projek en terugvoer van die onderwyser na die tyd.
- Die ontwikkeling van die projek is 'n deurlopende proses. Die onderwyser moet altyd kyk na die werk wat in die vorige fases gedoen is, wanneer 'n spesifieke fase van ontwikkeling geassesseer word. Die onderwyser moet die vordering van die projek, in verhouding met die werk wat in vorige fases gedoen is, deeglik monitor.
  - Die vereistes wat in Fase 1 geïdentifiseer en uitgespel is, moet in die ontwerpfase van die projek, d.w.s. Fase 2, gereflekteer word.
  - Die werk wat in die ontwerpfase, d.w.s. Fase 2, gedoen is, moet in die koderings- en implementeringsfase van die projek, d.w.s. Fase 3, gereflekteer word.
- Die leerder moet al die vorige dokumentasie (Fase 1- en Fase 2-dokumentasie) byderhand hê wanneer die demonstrasie as deel van Fase 3 plaasvind.
- Die demonstrasies moet elektronies op die rekenaar gedoen word.
- Die leerder moet sy/haar rekenaarprogram uitvoer en al die eienskappe van die program aan die onderwyser toon vir assessering.
- Die onderwyser moet die leerder vra om die toetsstrategieë uit te voer om alle fasette van die program te toets.
- Die onderwyser kan van die leerder verwag om ander, addisionele toetsprosedures uit te voer om seker te maak dat die hele program korrek werk.
- Die onderwyser moet die nasienblad vir Fase 3 gebruik om tydens die demonstrasie punte toe te ken.

- Die onderwyser moet **ewekansig 'n paar gedeeltes van die programmeringskode (uitsluitend die 10% geleende kode) in die projek identifiseer**. Geleende kode is enige programmingskode wat nie deur die standaardweergawe van Java of Delphi voorsien word nie en wat nie deur die leerder self geskryf is nie. **Die leerder moet dan die doel en die werking van die ewekansig-geselekteerde kode aan die onderwyser verduidelik**. Dit word gedoen om seker te maak dat leerders self die kode skryf. 'n Soortgelyke prosedure sal ook tydens eksterne moderering gevolg word. Indien die leerder nie die kode wat in die projek gebruik is, kan verduidelik nie, kan **geen punte vir die projek aan die leerder toegeken word nie**.
- Die leerder moet die elektroniese weergawe van die projek wat gedemonstreer is, inlewer. Die onderwyser sal hierdie weergawe gebruik om enige uitstaande punte toe te ken sodat die punt gefinaliseer kan word.

**LET WEL:** Sodra die produk van 'n fase ingehandig en geassesseer is, sal daardie fase nie weer geassesseer word nie.

## 6. Rekordhouding en rapportering

Vir elke fase sal die onderwyser die fase assesseer, die punt aanteken en terugvoer aan die leerder gee.

Die punte vir die verskillende fases sal bymekaar getel en verwerk word na 'n punt uit 100, wat die finale punt sal wees.

## 7. Riglyne vir die bestuur van die PAT

Daar is twee maniere waarop die PAT bestuur kan word:

### Opsie 1:

- Die onderwyser kan een of twee periodes per week aan die PAT afstaan terwyl hy/sy voortgaan met normale onderrig gedurende die res van die week om die graad 12-kurrikulum af te handel.
- As hy/sy hierdie opsie kies, moet hy/sy so gou as moontlik gedurende die eerste kwartaal met die PAT begin en een fase per kwartaal voltooi.

### Opsie 2:

- Die onderwyser kan 'n aaneenlopende tydperk aan die PAT afstaan, bv. die laaste gedeelte van elke kwartaal en een fase per kwartaal voltooi of eers die graad 12-kurrikulum afhandel en dan 'n aaneenlopende tydperk in die derde kwartaal gebruik.

Die onderwyser moet sy/haar werkskedule volgens die opsie wat hy/sy verkies, beplan.

Daar word aanbeveel dat die onderwyser die leerders se onderwerpe aanteken wanneer hulle met Fase 1 begin om te voorkom dat 'kitsprojekte', wat moontlik nie die leerder se eie werk is nie, kan opduik. Onderwysers moet seker maak dat leerders projekte 'registreer' wat hulle in staat is om te voltooi, om sodoende te voorkom dat onderwerpe verander word omdat die aanvanklike onderwerp te moeilik of tydrowend is.

## 8. WENKE

- Verduidelik eers die PAT en gee 'n oorsig oor die proses van ontwikkeling aan die leerders voordat hulle met Fase 1 begin.
- Bespreek die taak en die onderwerp met die leerders. Laat hulle toe om vrae te vra en maak seker dat hulle die probleem wat opgelos moet word, duidelik verstaan.
- Bespreek voorbeelde van moontlike toepassings binne die gegewe scenario met die leerders. Laat die leerders self met 'n paar idees vorendag kom en bespreek die geskiktheid daarvan.

- Alhoewel 'n ander scenario gebruik is, kan dit 'n goeie idee wees om vir die leerders goeie en swak voorbeelde van PAT'e van vorige jare te wys.
- Wees baie streng met die inlewering van die dokumentasie wat in elke fase vereis word.



**Inligtingstechnologie**  
**Praktiese Assesseringstaak (PAT)**  
**2012**

**Riglyne aan die leerder**

**Hierdie afdeling bestaan uit 10 bladsye.**

# Inligtingstegnologie – Praktiese Assesseringstaak (PAT)

Die IT-PAT bestaan uit drie fases:

- Ondersoek en analise
- Ontwerp
- Kodering, implementering en interne dokumentasie, insluitend aanbieding en algemene evaluering

## 1. Beplanning van die programmeringsprojek

### 1.1 Beskrywing

#### Scenario vir die 2012-graad 12-PAT:

Jy moet 'n program te skryf wat as 'n inligtingstelsel deur die personeel en/of besoekers van enige natuurgeoriënteerde toeriste-aantreklikheid in Suid-Afrika gebruik kan word, byvoorbeeld:

- 'n Natuurreserveaat
- 'n Wildreserveaat
- 'n Akwarium
- 'n Voëlpark
- 'n Dieretuin
- Enige ander wildlewe- of natuurgeoriënteerde toeriste-aantreklikheid in Suid-Afrika

Die fokus van die stelsel moet wees om natuurverwante inligting aan personeel en/of besoekers te verskaf.

**LET WEL:** Dit moet **NIE** 'n besprekingstelsel wees **NIE**.

Jy moet navorsing doen oor die inligtingsbehoefte van die personeel en/of besoekers van die toeriste-aantreklikheid van jou keuse. 'n Funksionele en bruikbare stelsel moet ontwerp word wat suksesvol by die spesifieke toeriste-aantreklikheid geïmplementeer kan word.

Jy kan jou eie onderwerp/toepassing binne die konteks van die gegewe scenario kies.

Met die voltooiing van die projek moet jy die volgende vaardighede toepas:

- Ondersoek
- Analise
- Ontwerp
- Ontwikkeling van programmatuur
  - Programmeringsvaardighede deur die gebruik van die programmeringstaal wat bestudeer is
  - Ontwikkeling van 'n databasis
  - Ontwerp van 'n grafiese gebruikerskoppelvlak (GUI)

Gedeeltes van die assesseringsinstrument sal projekte penaliseer wat nie aan die gegewe scenario verwant is nie.

**LET WEL:** Jou finale program moet uit EEN enkele, logies-verwante deel programmatuur bestaan. Projekte wat uit twee of meer nieverwante programme bestaan, sal slegs vir een van die gedeeltes punte kry omdat slegs een van die programme as die werklike projek beskou word.

## 1.2 Ontwikkelingsfases van die projek

Die programmeringsprojek moet in drie fases voltooi word, soos aangedui in die volgende tabel:

FASES	PUNTE	%
Fase 1: Onderzoek en analise	30	17,5%
Fase 2: Ontwerp	30	17,5%
Fase 3: Kodering, implementering en interne dokumentasie, insluitend aanbieding en evaluering	110	65%
<b>TOTAAL</b>	<b>170</b>	<b>100</b>

Dokumentasie/Bewys van wat jy gedurende elke fase van ontwikkeling gedoen het, moet volgens gespesifiseerde intervale ingelewer word. Die bewys en afvoer van elke fase word hieronder bespreek. Punte kan slegs toegeken word as jy hierdie bewyse aan die onderwyser voorsien **en** indien die werk wat geassesseer word, verband hou met die werk wat in vorige fases gedoen is.

**Datums vir die inlewing van die dokumentasie/bewys van die werk wat gedurende elke fase gedoen is, sal deur die onderwyser vasgestel word.**

Bestudeer die assesseringsinstrument vooraf om seker te maak dat jy aandag gee aan al die relevante vereistes volgens die assesseringsinstrumente.

Neem die terugvoer van die onderwyser, aangedui op die assesseringsinstrument, in ag en verbeter jou werk vir die volgende fase daarvolgens. Op 'n paar plekke word punte toegeken vir die verbetering van werk wat in vorige fases nie reg gedoen is nie.

Al die dokumentasie van die vorige fases moet tydens elke assessering vir die onderwyser beskikbaar wees.

## 1.3 Hulpbronne wat vir die projek benodig word

Jy sal die volgende hulpbronne nodig hê om die projek te voltooi:

- Toegang tot 'n rekenaar met die volgende programme:
  - Programmeringstaal: Java of Delphi
  - Kantoortoepassingspakket, soos 'n woordverwerker (bv. MS Word) en 'n aanbiedingspakket (bv. MS Powerpoint)
  - Databasisprogrammatuur, soos MS Access of MySQL
- IDE (vir Delphi is dit deel van die programmeringstaal, maar vir Java het jy addisionele programmatuur nodig, soos Netbeans)

Die projek moet onder **gekontroleerde toestande** voltooi word en deur die onderwyser gefasiliteer en deurlopend gemonitor word.

Jy moet aandag gee aan die volgende minimum kriteria:

### Fase 1: Onderzoek en analise

- Beskrywing van die probleem in jou eie woorde waarin die hoofaspekte in een paragraaf uitgelig word
- Onderzoek van die onderwerp deur:
  - Identifisering van die aard en die omvang van die probleem en insameling van feite van potensiele gebruikers, insluitend hulle behoeftes en enige beperkinge wat hulle mag hê
  - Verkryging van inligting vanuit 'n werklike situasie waarin die programmatuurprodukt deur die eindgebruiker gebruik gaan word

- Analisering van die probleem: Wat is die vereistes en wat moet deur die programmeringsoplossing voorsien word?

(Sien die volgende afdeling Fase 1 en Assesseringsinstrument Fase 1)

### **Fase 2: Ontwerp**

- Ontwerp 'n oplossing: Hoe sal die program/stelsel aan die vereistes voldoen? Voorsien 'n goed beplande oplossing in terme van:
  - Toevoer, verwerking en afvoer
  - Struktuur en inhoud van die databasis
  - GUI en die vloei van gebeure

(Sien die volgende afdeling Fase 2 en Assesseringsinstrument Fase 2)

### **Fase 3: Kodering, implementering en interne dokumentering**

- Die projek moet die belangrikste ontwikkelingsinstrumente insluit, m.a.w. 'n databasisonwerp en programmering wat toepaslik geïntegreer is. (Ander toepassings kan saam met hierdie ontwikkelingsinstrumente geïntegreer word.)
- Ander aspekte van die programmeringsprojek wat geassesseer sal word, sluit in:
  - Programmeringstyl
  - Grafiese gebruikerskoppelvlak (GUI)
  - Gebruik van mens-rekenaar-interaksiebeginsels ('HCI')
  - Funksionaliteit van die program
  - Vlak van kundige programmering
  - Robuustheid van die program, insluitend die gebruik van defensiewe programmeringstegnieke
  - Of die projek ooreenstem met die oorspronklike mikpunte en doelwitte
- Interne dokumentering om dele van die program te verduidelik

### **Aanbieding en algemene evaluering**

- 'n Drukstuk van die bronkode, kommentaar ingesluit (interne dokumentering)
- Demonstrasie van die finale produk en ondervraging (deur die onderwyser).
- Die onderwyser sal die volgende evalueer
  - Jou tydbestuur – Het jy al die sperdatums nagekom?
  - Jou vermoë om dele van kode wat ewekansig gekies word, te verduidelik
  - Die geskiktheid van die oplossing binne die konteks van die scenario

(Sien die volgende afdeling Fase 3 en Assesseringsinstrument Fase 3)

## **2. Instruksies vir die fases van die programmeringsprojek**

Die instruksies vir die verskillende fases is soos volg:

## FASE 1:

### Onderzoek en Analise

Sperdatum: \_\_\_\_\_

Met die voltooiing van hierdie fase moet jy agtergrondinligting oor jou onderwerp opspoor en bepaal *wat* die program/stelsel moet doen en voorsien:

#### 1. Probleemstelling

- Identifiseer en beskryf/verduidelik in jou eie woorde wat die taak en die probleem is wat opgelos moet word. Hierdie beskrywing moet nie 'n beskrywing van enige rekenaarkode of die oplossing wees nie. Al wat vereis word, is 'n beskrywing van die probleem wat jy in die werklike konteks of situasie waarin dit geïdentifiseer is, ondersoek.
- Jou probleemstelling moet:
  - Duidelik sê wat die probleem behels (verskaf kort agtergrondinligting, m.a.w. hoekom jy die onderwerp gekies het)
  - Sê wat jy probeer doen/oplos en vir wie
  - Die aspekte beskryf wat opgelos behoort te word/verwagtinge van die programmatuur
  - Aandui wat die doel van die programmatuur sal wees

#### 2. Onderzoek

- Onderzoek die onderwerp om feite in te samel en om die aard en die omvang van die probleem vas te stel, bv. wat sal deel wees van die programmatuur en wat sal nie ingesluit word nie. Die doel is om die vereistes, behoeftes en beperkings vas te stel.
- Jy sal dalk die volgende wil uitvind:
  - Watter stelsels of prosesse word in die afwesigheid van programmatuur gebruik (bv. papiergebaseerde oplossings) en wat is die beperkinge/uitdagings hiermee
  - Die sleutelareas waaraan aandag gegee/wat opgelos gaan word
  - Inligting wat spesifiek met jou onderwerp verband hou en waarvoor jy 'n oplossing moet konstrueer (bv. Wat 'n inligtingskiosk is, watter inligting mense gewoonlik van so 'n kiosk wil hê, watter data jy nodig sal hê, ens.) en hoe of waar/van wie jy hierdie inligting kan bekom
- Demografiese inligting oor gebruikers van jou stelsel (ouderdom, geslag, gestremdhede, vlak van geletterdheid, ens.)
- Jy moet 'n volledige lys van verwysings byhou van al die hulpbronne (webblaaie, boeke) wat gebruik is.
- Bewys van navorsing: Goed geformateerde en voorgestelde opsomming van al die bevindinge van die navorsing

#### 3. Bepaal vereistes vir die program

- Hou besprekings met die eindgebruikers, maak notas, versamel dokumente en maak opsommings en afleidings uit al hierdie inligting.
- Neem waar hoe die stelsel of 'n soortgelyke stelsel tans werk. Indien moontlik, maak notas.
- Maak notas oor toevoer, verwerking en afvoer wat vereis sal word.
- Moontlike bewyse wat ingelewer kan word: Vraelyste, notas (of opnames) wat gedurende onderhoude gemaak is, foto's, dokumente/vorms wat in die huidige stelsel gebruik word, oudio-/video-opnames van onderhoude/hoe die stelsel tans werk, ens.
- Inligting moet verskaf word oor die besonderhede van die onderhoude, vraelyste, gebruikers, ens. Inligting soos die tyd en datum wanneer die vraelys/onderhoud gedoen is, asook die name, plek en kontakbesonderhede van deelnemers. Jou onderwyser kan hierdie inligting gebruik om te verifieer dat die ondersoek gedoen is soos wat aangedui is.

## 4. Moontlike oplossing

- Gebruik jou navorsing en die terugvoer van die gebruikers en skryf 'n moontlike/ beoogde bondige oplossing vir die probleem in jou eie woorde.
- Moontlike bewyse om ingelewer te word: Definisie van die probleem en/of lys van vereistes. Dit is nie 'n gedetailleerde spesifikasie met toevoer en afvoer nie, maar eerder slegs 'n eenvoudige lys eienskappe wat die beoogde oplossing aan gebruikers sal voorsien.

## 5. Lewer 'n beplanningsdokument in

Kombineer die resultate van jou navorsing in 'n beplanningsdokument wat uit die volgende bestaan:

- Beskrywing van die probleem
- Agtergrondinligting oor die onderwerp met verwysings
- Bewyse van inligting wat ingesamel is
- Breë oorsig van 'n moontlike oplossing vir die probleem

## FASE 2:

### Ontwerp

Sperdatum: \_\_\_\_\_

Met die voltooiing van hierdie fase moet jy die spesifikasies van die program/stelsel bepaal en aandui *hoe* die program/stelsel die doelwitte sal bereik wat tydens die analise daargestel is. **Bestudeer die assesseringsinstrument vir Fase 2 en 3 om seker te maak dat jou projek voldoen aan die vereistes waaraan alle projekte moet voldoen.**

1. Skakel vereistes/eienskappe om in spesifikasies:  
Spesifiseer die oplossing in terme van:
  - Toevoer
  - Verwerking
  - Afvoer (tabelle, grafieke, geldeenheid, eenhede van meting, ens. ingesluit)Sluit die volgende in as deel van die spesifikasies:
  - Defensiewe programmeringstegnieke soos:
    - Datavalidering (dui aan waar en hoe dit toegepas moet word)
    - Foutboodskappe
  - Datastrukture om data te organiseer en te stoor:
    - Databasisontwerp
      - Velde (tipes, name en groottes)
      - Verwantskappe
      - Sleutels (primêre en vreemde)
      - ER-diagramme
    - Datatipes en strukture wat in programmering gebruik gaan word:
      - Primitiewe datatipes – enkele waarde, bv. 'integer', 'real/double'
      - Gestruktureerde datatipes – versamelings van data, bv. skikkings, matrikse, rekords, stelle ('sets'), kombinasielyste ('combo boxes'), lyste, vektore
      - Abstrakte datatipes – versamelings van stel data en stel bewerkings wat op die data uitgevoer kan word, bv. klasse en objekte
      - Ontwerp van klasse – metodes met parameters en terugstuur-datatipes, attribute, konstruktors, klasdiagramme, oorerwingshiërargieë waar van toepassing (hoër orde)
  - Grafiese gebruikerskoppelvlak-ontwerp (GUI)  
Tekeninge en/of skermkopieë ('screen shots') van die GUI wat die volgende aandui:
    - Uitleg van die komponente
    - Navigering (bv. Hoe beweeg die gebruiker van een skerm na die volgende en terug?)
    - Vloei van gebeure (van een skerm na die volgende, van een gebeurtenis na die volgende)
    - Vloei van data tussen eenhede/modules (Watter data word tussen modules gestuur?)
    - Motivering vir die gebruik van toevoer- en afvoerkomponente (bv. Hoekom is 'n kombinasielys ('combo box') gebruik in plaas van 'n 'text field', of hoekom is 'n 'check box' gebruik in plaas van 'n 'radio button'?)

2. Lewer 'n dokument in met al die programspesifikasies soos hierbo aangedui. Die dokument moet die volgende insluit:
- Toevoer-, verwerkings- en afvoervereistes
  - Fout-toetsing en valideringsprosedures
  - Databasisontwerp
  - Lys van datastrukture wat gebruik is (insluitend klasdiagramme)
  - Ontwerp en uitleg van die GUI

Die dokument kan TVA-tabelle, vloedigramme en diagramme soos EV-diagramme, klasdiagramme insluit en skermafdrucke of enige ander geskikte instrumente gebruik wat 'n duidelike voorstelling van die stelsel sal gee, insluitend die nodige aantekeninge en beskrywings van spesifikasies.



## FASE 3:

### Kodering en Implementering

Sperdatum: \_\_\_\_\_

Met voltooiing van hierdie fase sal jy die oplossing moet kodeer met insluiting van die grafiese gebruikerskoppelvlak (GUI) soos wat dit in die vorige fase beplan is, datastrukture skep, die program ontfout en toets. **Bestudeer die assesseringsinstrument vir Fase 3 om seker te maak dat jou projek voldoen aan die vereistes waaraan alle projekte moet voldoen.**

Voorgestelde stappe om hierdie fase te voltooi:

1. Breek die oplossing, soos beskryf in Fase 2, in modules (bv. volgens opsies) op.
2. Kodeer/Skep die grafiese gebruikerskoppelvlak (GUI).
3. Skep die datastrukture om die data te organiseer en te stoor.
4. Kodering – Kodeer die programoplossing volgens vereistes en spesifikasies (toevoer, verwerking en afvoer) wat in die vorige fase opgestel is. Punte sal onder andere toegeken word vir die volgende:
  - Toepaslike toevoerstrategieë
  - Toegang tot die databasis
  - Toepaslike en effektiewe algoritmes
  - Goeie programmeringsbeginsels: hergebruik van kode, veranderlike name, kommentaar
  - Parameter-oordrag: Onafhanklikheid van modules
5. Hantering van foute: Maak seker dat toevoer gevalideer word en dat hantering van uitsonderings toegepas word waar nodig.
6. Toetsing/Ontfouting – Maak seker dat die program reg uitvoer en geen foute het nie.
  - Gebruik 'n toetsstrategie om te verseker dat die program/stelsel doen wat dit behoort te doen deur verskillende reekse data te gebruik, insluitend ekstreme/foutiewe toetsdata. Hou boek van hierdie data aangesien dit in jou dokumentering benodig sal word.
7. Interne dokumentering
8. Evalueer jou program/stelsel – Doen die program wat dit moet doen/voldoen dit aan die vereistes? Het jy goeie programmeringsbeginsels toegepas?
  - Hersien die programkode. Is goeie programmeringsbeginsels toegepas? Bestudeer die assesseringsinstrument vir Fase 3 deeglik.
  - Werk die program/stelsel reg?
  - Stem die program ooreen met wat in Fase 1 en 2 aangedui word?
  - Is die grafiese gebruikerskoppelvlak (GUI) goed ontwerp?

**LET WEL:** Jy word toegelaat om gebruik te maak van 'geleende' kode, maar dit mag nie meer as 10% van jou programmeringskode uitmaak nie. Geleende kode is enige programmeringskode wat nie deur die standaard weergawe van Java of Delphi voorsien word nie en wat nie deur die leerder self geskryf is nie. Dit sal tipies kode wees om uitsonderlike/ongewone funksies uit te voer soos om 'n video-uittreksel te speel, ens. **Jou program sal nie deur die onderwyser aanvaar word indien die 'geleende' kode die perk van 10% oorskry nie.**

## 9. Aanbieding en evaluering

### 1. Lewer in:

- Elektroniese kopie van programme en al die betrokke lêers, bv. datalêers, ens.

### 2. Demonstreer die program vir evaluering en assessering

#### Riglyne vir die demonstrasie van die projek:

- Die onderwyser sal datums en tye skeduleer vir demonstrasies. Ongeveer 20 minute sal per projek toegelaat word.
- Jy moet al die dokumentasie ten minste EEN week voordat die demonstrasie plaasvind, inlewer.
- Die demonstrasie moet elektronies op 'n rekenaar gedoen word.
- Jy moet jou program op die rekenaar uitvoer en al die funksies wat die program bied, aan die onderwyser uitwys vir evaluering.
- Die onderwyser kan 'n toetsstrategie as 'n riglyn gebruik en die leerder vra om sekere of al die toetsprosedures uit te voer.
- Die onderwyser kan jou vra om addisionele toetsprosedures uit te voer om seker te maak dat die hele program korrek werk.
- Die onderwyser kan die nasienblad vir Fase 3 as riglyn gebruik en punte daarvolgens toeken tydens die demonstrasie.
- Die onderwyser moet willekeurig 'n paar gedeeltes programmeringskode (uitsluitend die 10% geleende kode) in die projek uitwys. Jy moet dan aan die onderwyser die doel en werking van die willekeurig gekose kode aan die onderwyser verduidelik. Dit word gedoen om seker te maak dat jy alle kode self geskryf het. 'n Soortgelyke prosedure sal ook tydens eksterne moderering gevolg word. Indien jy nie die kode wat in jou eie projek gebruik is, kan verduidelik nie, kan geen punte vir die projek toegeken word nie.
- Jy moet na afloop van die demonstrasie die elektroniese weergawe van die projek inlewer. Die onderwyser sal hierdie weergawe gebruik om enige uitstaande punte toe te ken sodat die punt gefinaliseer kan word.

### 3. Finale algemene evaluering

Die onderwyser sal die volgende evalueer:

- Tydbestuur – Het jy al die sperdatums nagekom?
- Gebruikswaarde – Is jou oplossing van toepassing binne die konteks van die gegewe scenario?
- Hou die verskillende fases van ontwikkeling verband met mekaar en lei dit tot die finale oplossing as een aaneenlopende proses?

**Inligtingstechnologie**

**Praktiese Assesseringstaak (PAT)**

**2012**

**Assesseringsinstrumente**

**Hierdie afdeling bestaan uit 10 bladsye.**

# Assesseringsinstrumente vir die programmeringsprojek

Assessering vir Fase 1

Naam van leerder: \_\_\_\_\_

Onderzoek en Analise: Kriteria					Moontlike Punt	Punt	Behaal		
Probleemstelling	Die probleemstelling					4			
	4	3	2	1	0				
	Die probleem is duidelik gestel en beskryf en ondubbelsinnig – stel duidelik wat die probleem behels; Lê uit wat hy/sy probeer doen en watter aspekte opgelos moet word; Duidelike stelling van wat die doel van die programmatuur sal wees	Die stelling is duidelik, maar het geringe tekortkominge.	Die stelling is vaag en laat die leser onseker oor wat die doel van die stelsel is.	Die stelling is so vaag dat geen waarneembare doel gevind kan word nie.	Geen stelling of beskrywing nie				
	Huidige stelsels/prosesse							2	
	2	1	0						
Duidelike beskrywing van huidige stelsel/prosesse, insluitend beperkinge en uitdagings en hoe die program hierdie kan aanspreek	Geringe tekortkominge; Beskrywing nie altyd duidelik nie	Geen beskrywing van huidige stelsels/prosesse of beperkinge en uitdagings nie							
Onderzoek	Sleutelareas met betrekking tot die probleem					4			
	4	3	2	1	0				
	Onderzoek definieer/ verduidelik omvattend en duidelik die sleutelareas rakende die onderwerp; Toon goeie insig en begrip van die sleutelareas van die onderwerp	Geringe tekortkominge; Toon insig in die meeste van die sleutelareas van die onderwerp	Toon 'n mate van insig in sommige sleutelareas	Vaag; Toon min begrip van sleutelareas	Sleutelareas nie gedefinieer nie				
	Lys van verwysings								
2	1	0							
Omvattende lys; In aanvaarbare formaat	Minder omvattend – geringe tekortkominge	Geen verwysings nie	2						

## NSS – Assesseringsinstrumente

Vereistes	<b>Toevoervereistes</b>				3		
	<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>			
	Omvattende lys van toevoervereistes verkry van gebruikers, goed gedefinieerd en verkry deur 'n geskikte metode, soos vraelyste, onderhoude, te gebruik	Geringe tekortkominge; Minder omvattende lys van toevoervereistes verkry van gebruikers; Nie so duidelik gedefinieerd nie, maar nog steeds aanvaarbaar	Min toevoervereistes verkry van verbruikers of vereistes is opgemaak deur die leerder; Vaag en nie duidelik gedefinieerd nie	Geen toevoervereistes verkry nie			
	<b>Verwerkingsvereistes</b>						
	<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>	3		
	Omvattende lys van verwerkingvereistes verkry van gebruikers, goed gedefinieerd en verkry deur 'n geskikte metode, soos 'n vraelys, onderhoud, te gebruik	Minder omvattende lys van verwerkingvereistes verkry van gebruikers deur ten minste een metode te gebruik; Nie so duidelik gedefinieerd nie, maar nog steeds aanvaarbaar	Min verwerkingvereistes verkry van gebruikers of vereistes is opgemaak deur die leerder; Vaag en nie duidelik gedefinieerd nie	Geen verwerkingvereistes verkry nie			
	<b>Afvoervereistes</b>						
	<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>			
	Omvattende lys van afvoervereistes verkry van gebruikers, goed gedefinieerd en verkry deur 'n geskikte metode, soos 'n vraelys, onderhoud, te gebruik	Minder omvattende lys van afvoervereistes verkry van gebruikers deur ten minste een metode te gebruik; Nie so duidelik gedefinieerd nie, maar nog steeds aanvaarbaar	Min afvoervereistes verkry van verbruikers of vereistes is opgemaak deur die leerder; Vaag en nie duidelik gedefinieerd nie	Geen afvoervereistes verkry nie	3		
	<b>Moontlike oplossing</b>						
	<b>4</b>	<b>3</b>	<b>2</b>	<b>1</b>			<b>0</b>
	Duidelike en omvattende beskrywing – 'n duidelike oorsig van 'n moontlike oplossing is gegee; Dui duidelik aan wat die program sal insluit en doen	Duidelik, maar met geringe tekortkominge	Basiese beskrywing is gegee, maar sommige aspekte van die voorgestelde oplossing is vaag	Die beskrywing is vaag; Geen duidelike oplossing geïdentifiseer in die beskrywing nie			Geen moontlike oplossing gegee nie
Algemeen	<b>Die reikwydte (grense en aannames/eienskappe wat ingesluit sal word)</b>				3		
	<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>			
	Die omvang van die voorgestelde oplossing is duidelik en goed gedefinieerd; Toon duidelik aan wat deel sal wees van die program en wat dit nie sal insluit nie	Gedefinieerd, maar sommige geringe tekortkominge; Nie altyd duidelik oor wat die program sal insluit of nie sal insluit nie	Vaag gedefinieerd; Geen duidelike reikwydte kan bepaal word nie	Nie gedefinieerd nie			
	<b>Toepaslikheid van voorgestelde oplossing in die konteks van die scenario</b>						
	<b>2</b>	<b>1</b>	<b>0</b>		2		
	Mees toepaslik; Goeie toepassing in die scenario	Toepaslik; Toepassing in scenario nie altyd oortuigend nie	Nie toepaslik nie; Toepassing in scenario nie oortuigend nie				
<b>Totaal:</b>					<b>30</b>		

## Assessering vir Fase 2:

Naam van leerder: \_\_\_\_\_

<b>Ontwerp: Kriteria</b>				Moontlike Punt	Punt Behaal					
<b>Spesifikasies</b>	<b>Toevoer</b>				3					
	<table border="1" style="width: 100%; text-align: center;"> <tr> <td><b>3</b></td> <td><b>2</b></td> <td><b>1</b></td> <td><b>0</b></td> </tr> </table>						<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>
	<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>						
	<table border="1" style="width: 100%;"> <tr> <td style="width: 25%;">Gebruikerstoevoer en ander bronne van toevoer duidelik beskryf in terme van watter toevoer vereis word asook die formaat (bv. datumformaat) van die toevoer</td> <td style="width: 25%;">Beskryf met geringe tekortkominge in terme van wat vereis word en in watter formaat</td> <td style="width: 25%;">Beskrywing vaag of onvolledig</td> <td style="width: 25%;">Slegs gelys – geen beskrywing nie</td> </tr> </table>				Gebruikerstoevoer en ander bronne van toevoer duidelik beskryf in terme van watter toevoer vereis word asook die formaat (bv. datumformaat) van die toevoer	Beskryf met geringe tekortkominge in terme van wat vereis word en in watter formaat	Beskrywing vaag of onvolledig	Slegs gelys – geen beskrywing nie		
	Gebruikerstoevoer en ander bronne van toevoer duidelik beskryf in terme van watter toevoer vereis word asook die formaat (bv. datumformaat) van die toevoer	Beskryf met geringe tekortkominge in terme van wat vereis word en in watter formaat	Beskrywing vaag of onvolledig	Slegs gelys – geen beskrywing nie						
	<b>Verwerking</b>				3					
	<table border="1" style="width: 100%; text-align: center;"> <tr> <td><b>3</b></td> <td><b>2</b></td> <td><b>1</b></td> <td><b>0</b></td> </tr> </table>						<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>
	<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>						
	<table border="1" style="width: 100%;"> <tr> <td style="width: 25%;">Beskryf duidelik hoe data verwerk/ gemanipuleer sal word in terme van vereistes, formaat, berekeninge, formules, ens.; Kort, duidelike en korrekte beskrywing in alle gevalle waar toepaslik; Gebruiker het duidelikheid oor resultaat</td> <td style="width: 25%;">Beskrywing van verwerking/ manipulering van data is gedoen/duidelik in die meeste van die gevalle waar dit vereis word; Goeie poging, maar kan verbeter; Nie altyd korrek of toepaslik nie</td> <td style="width: 25%;">Beskrywing van verwerking/ manipulering van data nie duidelik in die meeste gevalle nie; Beskrywing is vaag/ onvolledig; Nie korrek/toepaslik in die meeste gevalle nie</td> <td style="width: 25%;">Verwerking/manipulering van data nie beskryf nie</td> </tr> </table>				Beskryf duidelik hoe data verwerk/ gemanipuleer sal word in terme van vereistes, formaat, berekeninge, formules, ens.; Kort, duidelike en korrekte beskrywing in alle gevalle waar toepaslik; Gebruiker het duidelikheid oor resultaat	Beskrywing van verwerking/ manipulering van data is gedoen/duidelik in die meeste van die gevalle waar dit vereis word; Goeie poging, maar kan verbeter; Nie altyd korrek of toepaslik nie	Beskrywing van verwerking/ manipulering van data nie duidelik in die meeste gevalle nie; Beskrywing is vaag/ onvolledig; Nie korrek/toepaslik in die meeste gevalle nie	Verwerking/manipulering van data nie beskryf nie		
	Beskryf duidelik hoe data verwerk/ gemanipuleer sal word in terme van vereistes, formaat, berekeninge, formules, ens.; Kort, duidelike en korrekte beskrywing in alle gevalle waar toepaslik; Gebruiker het duidelikheid oor resultaat	Beskrywing van verwerking/ manipulering van data is gedoen/duidelik in die meeste van die gevalle waar dit vereis word; Goeie poging, maar kan verbeter; Nie altyd korrek of toepaslik nie	Beskrywing van verwerking/ manipulering van data nie duidelik in die meeste gevalle nie; Beskrywing is vaag/ onvolledig; Nie korrek/toepaslik in die meeste gevalle nie	Verwerking/manipulering van data nie beskryf nie						
	<b>Afvoer</b>				3					
	<table border="1" style="width: 100%; text-align: center;"> <tr> <td><b>3</b></td> <td><b>2</b></td> <td><b>1</b></td> <td><b>0</b></td> </tr> </table>						<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>
<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>							
<table border="1" style="width: 100%;"> <tr> <td style="width: 25%;">Vereiste afvoer (skerm, verslag) geïdentifiseer en duidelik beskryf volgens die vereistes en in terme van formaat (geldeenheid, meeteenheid, ens.)</td> <td style="width: 25%;">Geïdentifiseer en beskryf met geringe tekortkominge</td> <td style="width: 25%;">Die beskrywing is vaag of die meeste van die vereiste afvoer is nie geïdentifiseer nie of die formaat van afvoer is nie beskryf nie</td> <td style="width: 25%;">Afvoer nie geïdentifiseer en beskryf nie</td> </tr> </table>				Vereiste afvoer (skerm, verslag) geïdentifiseer en duidelik beskryf volgens die vereistes en in terme van formaat (geldeenheid, meeteenheid, ens.)	Geïdentifiseer en beskryf met geringe tekortkominge	Die beskrywing is vaag of die meeste van die vereiste afvoer is nie geïdentifiseer nie of die formaat van afvoer is nie beskryf nie	Afvoer nie geïdentifiseer en beskryf nie			
Vereiste afvoer (skerm, verslag) geïdentifiseer en duidelik beskryf volgens die vereistes en in terme van formaat (geldeenheid, meeteenheid, ens.)	Geïdentifiseer en beskryf met geringe tekortkominge	Die beskrywing is vaag of die meeste van die vereiste afvoer is nie geïdentifiseer nie of die formaat van afvoer is nie beskryf nie	Afvoer nie geïdentifiseer en beskryf nie							
<b>Datavalidering</b>				3						
<table border="1" style="width: 100%; text-align: center;"> <tr> <td><b>3</b></td> <td><b>2</b></td> <td><b>1</b></td> <td><b>0</b></td> </tr> </table>						<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>	
<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>							
<table border="1" style="width: 100%;"> <tr> <td style="width: 25%;">Aangedui vir alle toevoer en in detail beskryf; Betekenisvol en effektief</td> <td style="width: 25%;">Geïdentifiseer en beskryf met geringe tekortkominge</td> <td style="width: 25%;">In sommige gevalle aangedui waar van toepassing en/of die beskrywing van validering is vaag of onvolledig</td> <td style="width: 25%;">Nie aangedui nie</td> </tr> </table>				Aangedui vir alle toevoer en in detail beskryf; Betekenisvol en effektief	Geïdentifiseer en beskryf met geringe tekortkominge	In sommige gevalle aangedui waar van toepassing en/of die beskrywing van validering is vaag of onvolledig	Nie aangedui nie			
Aangedui vir alle toevoer en in detail beskryf; Betekenisvol en effektief	Geïdentifiseer en beskryf met geringe tekortkominge	In sommige gevalle aangedui waar van toepassing en/of die beskrywing van validering is vaag of onvolledig	Nie aangedui nie							
<b>Foutboodskappe geassosieer met datavalidering</b>				2						
<table border="1" style="width: 100%; text-align: center;"> <tr> <td><b>2</b></td> <td><b>1</b></td> <td><b>0</b></td> </tr> </table>						<b>2</b>	<b>1</b>	<b>0</b>		
<b>2</b>	<b>1</b>	<b>0</b>								
<table border="1" style="width: 100%;"> <tr> <td style="width: 33%;">Aangedui vir alle toepaslike valideringskontrole/-foute en in detail beskryf/aangetoon</td> <td style="width: 33%;">Sommige foutboodskappe aangedui en/of beskrywing van foutboodskappe is vaag</td> <td style="width: 33%;">Geen foutboodskappe aangedui vir enige datavalidering of foute nie</td> </tr> </table>				Aangedui vir alle toepaslike valideringskontrole/-foute en in detail beskryf/aangetoon	Sommige foutboodskappe aangedui en/of beskrywing van foutboodskappe is vaag	Geen foutboodskappe aangedui vir enige datavalidering of foute nie				
Aangedui vir alle toepaslike valideringskontrole/-foute en in detail beskryf/aangetoon	Sommige foutboodskappe aangedui en/of beskrywing van foutboodskappe is vaag	Geen foutboodskappe aangedui vir enige datavalidering of foute nie								

	<b>Velde</b>				2					
	2		1				0			
	Goed gekose velde, veldtipes en -groottes om die toepassing te pas; Veldeienskappe soos validering en maskers aangetoon waar van toepassing		Meestal goed gekose velde, veldtipes en -groottes om die toepassing te pas; Veldeienskappe soos validering en maskers nie altyd aangedui waar van toepassing nie				Velde, veldtipes en -groottes nie goed gekies om toepassing te pas nie; Veldeienskappe soos validering en maskers nie aangedui nie			
Oorsig van databasisontwerp	<b>Tabelle</b>				3					
	3		2				1		0	
	Velde goed gegropeer in tabelle; Geen herhaling van velde in tabelle nie; Meer as twee tabelle met korrekte verwantskappe tussen die tabelle; Toon korrekte primêre en vreemde sleutels		Velde goed gegropeer in tabelle met geringe tekortkominge; Herhaling van velde in tabelle in sommige gevalle; Ten minste twee tabelle met korrekte verwantskappe tussen tabelle; Toon korrekte primêre en vreemde sleutels				Velde meestal goed gegropeer in tabelle; Uitgebreide herhaling van velde in tabelle; Ten minste twee tabelle maar geen/verkeerde verwantskap tussen tabelle nie; Toon geen/verkeerde primêre en vreemde sleutels		Slegs een tabel/slegs velde gelys	
Datastrukture	<b>Organiseer en stoor data</b> (versamelings met stel data en stel bewerkings wat op data uitgevoer kan word, bv. klasse en objekte)				3					
	3		2				1		0	
	Abstrakte datatipes gebruik; Toepassing goed beplan in terme van klasse en objekte; Objekte goed gestruktureer met toepaslike metodes		Gebruik abstrakte datatipes – ruimte vir verbetering; Toepassing nie heeltemal beplan in terme van klasse en objekte nie; Nie altyd van toepassing nie; Hele program nie objek-georiënteerd nie				'n Poging om abstrakte datatipes te gebruik; Objekte nie altyd goed saamgestel nie; Objekte saamgestel, maar nie korrek gebruik nie/glad nie gebruik nie; Baie min van die program is objek-georiënteerd		Geen abstrakte datatipes nie (klasse en objekte)	
GUI-ontwerp	<b>Toevoer</b>				3					
	3		2				1		0	
	Toepaslike toevoerkomponente om akkurate/geldige toevoer in alle gevalle waar vereis, te verseker; Gemotiveerde keuses waar vereis word		Toepaslike komponente waar vereis word; Keuses meestal gemotiveer; Geringe tekortkominge				Toepaslik in die meeste gevalle waar vereis word; Keuses nie altyd gemotiveer nie		Nie gedoen nie/nie toepaslike toevoerkomponente geïdentifiseer nie	
GUI-ontwerp	<b>Afvoer</b>				3					
	3		2				1		0	
	Toepaslike afvoerkomponente geïdentifiseer in alle gevalle waar vereis;		Toepaslike afvoerkomponente geïdentifiseer waar vereis – geringe tekortkominge				Toepaslik in die meeste gevalle waar vereis word; Keuses nie altyd gemotiveer nie		Nie gedoen nie/nie toepaslike afvoerkomponente geïdentifiseer nie	
Algemeen	<b>Spreek die vereistes gespesifiseer in Fase 1 aan</b>				2					
	2		1				0			
	Aandag gegee aan alle vereistes		Ten minste 50%				Minder as 50%			
<b>Totaal:</b>					<b>30</b>					

Assesering vir Fase 3:

Naam van leerder: \_\_\_\_\_

**LET WEL:** Fase 3 en 4 sal slegs geassesseer word wanneer Fase 1 and 2 voltooi is EN Fase 1 en 2 verband hou met die onderwerp gedek in Fase 3 en 4.

Kodering Implementering: Kriteria					Moontlike Punt	Punt	Behaal	
Databasis	<b>Databasistabelle</b>				4			
	4	3	2	1				0
	Aantal tabelle voor-sien 'n effektiewe oplossing – tabelle is genormaliseer (toepaslike getal tabelle om 'n doeltreffende oplossing te bied); Alle verwantskappe is goed gedefinieer deur gebruik van korrekte primêre en vreemde sleutels; Primêre en vreemde sleutels in verwante tabelle effektief en toepaslik	Getal tabelle bied 'n goeie oplossing; Ten minste een doeltreffende en toepaslike verwantskap wat die korrekte primêre of vreemde sleutels gebruik; Sommige primêre of vreemde sleutels in verwante tabelle nie toepaslik nie	Getal tabelle bied 'n gedeeltelike oplossing – tabelle nie genormaliseer nie (kon meer tabelle gebruik het vir 'n meer effektiewe oplossing); Die meeste van die primêre of vreemde sleutels nie geskik nie of slegs versteksleutels gebruik	Getal tabelle nie relevant tot die oplossing nie (slegs een tabel/'n aantal irrelevante tabelle met herhaling van velde in die tabelle; of meer as een tabel in elke databasis in plaas van een databasis)				Geen databasis gebruik nie
	<b>Veldtipes</b>							
	2	1	0					
	Alle velde met toepaslike datatipes	Datatipes van sommige velde nie toepaslik nie	Slegs verstektipes gebruik of geen databasis gebruik nie		2			
	<b>Veldgroottes - databasis</b>				2			
	2	1	0					
	Alle velde met toepaslike veldgroottes	Sommige van die velde is te groot/ klein	Slegs verstek-veldgroottes gebruik of geen databasis gebruik nie					
	<b>Beskrywende veldname in databasis</b>				1			
<b>LET WEL: Asseser die kode indien die program nie uitvoer nie. Hierdie afdeling het te make met die programmering in Delphi/Java, uitgesluit SQL.</b>								
Programmeringsoplossing	<b>Gebruikersgedefinieerde datastrukture (uitgesluit databasistabelle)</b>				3			
	3	2	1	0				
	Toepaslike en mees doeltreffende datastrukture in alle gevalle gebruik om die probleem op te los	Toepaslike en mees doeltreffende datastrukture in die meeste gevalle gebruik om die probleem op te los met geringe tekortkominge	Toepaslike en mees doeltreffende gebruik van datastrukture in minder as 50% van gevalle	Ontoepaslike of ondoeltreffende gebruik van datastrukture				
	<b>Veranderlikes/datastrukture se name betekenisvol</b>				2			
	2	1	0					
Veranderlikes/datastrukture se name is betekenisvol in alle gevalle, regdeur die program	Geringe tekortkominge	Nie betekenisvol nie						
<b>Seleksiestrukture</b>				3				
3	2	1	0					
Toepaslike en mees doeltreffende keusestrukture in alle gevalle gebruik om die probleem op te los	Toepaslike en mees doeltreffende keusestrukture in die meeste gevalle gebruik	Onvanpaste of ondoeltreffende gebruik van keusestrukture in die meeste gevalle	Nie gebruik nie					



<b>Herhalingstrukture</b>				
<b>3</b>		<b>2</b>		<b>1</b>
Toepaslike en die mees doeltreffende herhalingstrukture in alle gevalle gebruik om die probleem op te los		Toepaslike en die mees doeltreffende herhalingstrukture in die meeste gevalle gebruik		Onvanpaste of ondoeltreffende gebruik van herhalingstrukture in die meeste gevalle
				Nie gebruik nie
3				
<b>Komplekse programmeringstegnieke (bv. speel videosnitte, geleende kode, rafels ('threads'), netwerk)</b>				
<b>3</b>		<b>2</b>		<b>1</b>
Dit werk korrek; Toepaslik gebruik en voeg waarde toe tot die oplossing		Dit werk korrek; Nie altyd toepaslik gebruik nie of voeg nie regtig waarde toe tot die oplossing nie		Poging aangewend, maar dit werk nie behoorlik nie; Ontoepaslik gebruik; Nie relevant vir die oplossing nie
				Geen komplekse tegnieke gebruik nie of oorskry meer as 10% van die kode
3				
<b>Datavloei en prosesse (gebruikersgedefinieerde parameteroordrag)</b>				
<b>3</b>		<b>2</b>		<b>1</b>
Uitstekende interaksie/kommunikasie tussen modules/klasse; Sluit gevorderde datatipes as parameters of terugstuurtipes in		Vaardige/voldoende/'n mate van kommunikasie tussen modules/klasse met kleiner foute; Sluit 'n mate van parameteroordrag tussen modules in		Beperte kommunikasie tussen modules/klasse; Slegs primitiewe datatipes as parameters oorgedra
				Geen kommunikasie tussen modules/klasse nie; Geen parameters oorgedra nie
3				
<b>Hergebruik van kode (klasse en metodes/subprogramme)</b>				
<b>3</b>		<b>2</b>		<b>1</b>
Toepaslike en doeltreffende hergebruik van kode en/of metodes; Modules onafhanklik; Lokale veranderlikes gebruik waar toepaslik; Globale veranderlikes slegs gebruik wanneer vereis		Herbruik van kode/of metodes, maar nie altyd toepaslik/effektief nie; Modules nie altyd onafhanklik nie; Sommige globale veranderlikes beperk onafhanklikheid van modules		Herbruik van kode nie toepaslik/effektief nie; Modules kon afgebreek gewees het in meer modules; Byna alle veranderlikes globaal verklaar
				Lineêre programmering – een aaneenlopende program, geen modules nie; Geen hergebruik van kode en/of metodes nie
3				
<b>Oplossingalgoritmes</b>				
<b>3</b>		<b>2</b>		<b>1</b>
Alle oplossingalgoritmes gebruik om probleem op te los is toepaslik en doeltreffend		Toepaslik en doeltreffend met geringe tekortkominge		Meestal toepaslik of nie doeltreffend nie
				Totaal ontoepaslike oplossingalgoritmes of meestal ondoeltreffend
3				
<b>Korrektheid van oplossingalgoritmes</b>				
<b>3</b>		<b>2</b>		<b>1</b>
Geen logiese foute nie; Alle verwerkingsresultate korrek		Geringe tekortkominge; Baie min logiese foute; Baie min van die resultate is nie korrek nie		Logiese foute; Baie resultate is foutief
				Baie logiese foute; Byna al die resultate is foutief/min van die vereiste resultate word gelewer
3				
<b>Toevoerstrategieë</b>				
<b>3</b>		<b>2</b>		<b>1</b>
Die mees toepaslike, doeltreffende toevoerstrategieë in alle gevalle, bv. toevoer van teksleërs, databasistabelle, gebruikerstoevoer		Toepaslik en doeltreffend met geringe tekortkominge		Sommige strategieë kon meer geskik/doeltreffend wees
				Meestal ontoepaslik of ondoeltreffend
3				

Afvoer	<b>Afvoer vs. vereistes</b>				3
	3	2	1	0	
	Afvoer voldoen aan al die vereistes vir die oplossing	Afvoer voldoen aan die meeste van die vereistes vir die oplossing	Afvoer voldoen aan minder as 50% van die vereistes vir die oplossing	Geen afvoer nie	
	<b>Struktuur van afvoer (kodering)</b>				
	3	2	1		3
Afvoer altyd goed gestruktureerd, leesbaar met opskrifte en subopskrifte; Opskrifte herhaal op volgende bladsye/skerm waar van toepassing	Afvoer in die meeste gevalle goed gestruktureerd, leesbaar met opskrifte en subopskrifte; Opskrifte herhaal in die meeste gevalle op volgende bladsye/skerm waar van toepassing	Afvoer nie goed gestruktureerd nie; Opskrifte en/of subopskrifte in die meeste gevalle nie goed geformuleer nie of afwesig; Opskrifte meestal nie op volgende bladsy herhaal nie, waar van toepassing	Geen kode om afvoer te vertoon nie		
<b>Formaat van afvoer – alle waardes toepaslik geformateer waar van toepassing, bv. geldeenhede, meeteenhede, ens.</b>				1	
Databasisinteraksie	<b>Databasiskoppeling – string/pad korrek opgestel en dit werk</b>				1
	<b>Databasisinteraksie met program</b>				2
	2	1	0		
	Databasisinteraksie met program betekenisvol, bv. navrae en verslae	Nie altyd betekenisvol nie	Geen interaksie/nie betekenisvol nie		
	<b>Rekords korrek gemanipuleer via SQL (2 punte vir elke korrekte SQL-stelling tot 'n maksimum van 6 punte)</b>				6
	'Insert', 'delete', 'select', 'update' Lys ander:				
<b>Manipuleer velde via SQL (2 punte vir elke korrekte SQL-stelling tot 'n maksimum van 6 punte)</b>				6	
Berekeninge op velde, verander inhoud, vertoon slegs een veld, benoemde velde, rangskik volgens velde Lys ander:					
Fouthantering en toetsing	<b>Program kompilleer suksesvol – geen sintaksfoute nie</b>				1
	<b>Uitvoerfoute</b>				3
	3	2	1	0	
	Geen uitvoerfoute nie. Alle opsies voer suksesvol uit.	Sommige van die opsies lewer foute wanneer dit uitgevoer word.	Slegs een of twee opsies kan suksesvol uitvoer	Geen opsies voer suksesvol uit nie.	
	<b>Toevoervalidering</b>				3
	3	2	1	0	
	Alle toevoer wat gevalideer behoort te word, word gevalideer deur gebruik van kode.	Die meeste van die toevoer wat gevalideer behoort te word, word gevalideer.	Minder as 50% van die toevoer wat gevalideer behoort te word, word gevalideer.	Geen validering nie	
	<b>Foutboodskappe</b>				3
	3	2	1	0	
	Toepaslike en gebruikersvriendelike foutboodskappe in alle gevalle waar datavalidering toegepas is	Toepaslike en gebruikersvriendelike foutboodskappe in die meeste gevalle waar datavalidering toegepas is	Toepaslike en gebruikersvriendelike foutboodskappe in enkele gevalle waar datavalidering toegepas is	Geen foutboodskappe nie	
<b>Program gee afvoer (afvoer is die resultaat van verwerking, m.a.w. GUI-vensters met geen funksionaliteit klassifiseer nie as afvoer nie)</b>				2	
<b>Korrektheid van afvoer</b>				3	
3	2	1	0		
Program gee korrekte en toepaslike afvoer in alle gevalle	Program gee korrekte en toepaslike afvoer in die meeste gevalle	Program gee korrekte en toepaslike afvoer in minder as 50% van gevalle	Geen afvoer nie of slegs foutiewe afvoer		
<b>Hantering van uitsonderings ('Exception handling')</b>				2	
2	1	0			
Gebruik 'try... catch'-hantering van uitsonderings	Poging aangewend om uitsonderings te hanteer of gebruik if-stellings om fouthantering te doen	Geen poging nie			

Grafiese gebruikerskoppelvlak (GUI)	<b>Verskillende skerms toepaslik gebruik (vensters/paneel/'tab sheets', ens.)</b>			1
	<b>Komponente vir toevoer/afvoer gebruik (geïmplementeer soos vir Fase 2) OF toepaslike veranderinge gemaak vanaf Fase 2 se terugvoer)</b>			2
	2	1	0	
	Altyd toepaslik, die doeltreffendste	Geringe tekortkominge	Meestal nie toepaslik nie/ ondoeltreffend	
	<b>Byskrifte/lokboodskappe met eksakte formate vir toevoer</b>			2
	2	1	0	
	Konstant toegepas regdeur die hele projek waar vereis word	Toegepas in die meeste gevalle waar vereis word	Meestal nie toegepas nie	
	<b>Konsekwente uitleg (dieselfde voorkoms en gevoel regdeur die program)</b>			2
	2	1	0	
	Vir alle skerms (dieselfde kleure, skrifstipes gebruik regdeur program)	Die meeste van die skerms (sommige skerms verskillende kleure, skrifstipes, ens.) – geringe tekortkominge	Uitleg meestal nie konsekwent nie	
	<b>Insiggewende afvoer/verslae is insiggewend en maklik om te lees en te interpreteer</b>			2
	2	1	0	
	Alle afvoer, alle skerms is insiggewend en maklik om te lees en te interpreteer (toepaslike skrifgrootte, uitleg, kleur, ens.)	Die meeste van die afvoer is insiggewend en maklik om te lees en te interpreteer – geringe tekortkominge	Afvoer meestal nie insiggewend of nie maklik om te lees en te interpreteer nie	
	<b>Groepering van toevoer/afvoer</b>			2
	2	1	0	
	Tipe toevoer/afvoer saam gegroepeer, bv. adresinligting, vir alle skerms	Tipe toevoer/afvoer meestal saam gegroepeer – geringe tekortkominge	In die meeste gevalle nie saam gegroepeer nie	
	<b>Navigasie tussen skerms</b>			2
	2	1	0	
	Maklik om tussen skerms te navigeer – logiese vloe van gebeure	Maklik en logies met geringe tekortkominge	Nie maklik of logies nie	
	<b>Hulp beskikbaar as deel van die program</b>			2
2	1	0		
Hulp beskikbaar as deel van die program, werk korrek met duidelike instruksies	Nie altyd beskikbaar of duidelik nie of werk nie altyd korrek nie	Nie beskikbaar nie		
<b>Konteks sensitief</b>			2	
2	1	0		
Konteks-sensitiewe hulp beskikbaar ('tool tip'-teks ingesluit), toepaslik en effektief	Nie altyd toepaslik of effektief nie	Nie beskikbaar nie		
<b>Ontwerp vs. teikengebruiker</b>			2	
2	1	0		
Ontwerp konsidereer teikengehoor (ouderdom, geletterdheid, beperkte visie, toepaslikheid van beelde, ens.); Ontwerp geskik vir teikengebruiker	Gekonsidereer, maar geringe tekortkominge	Heeltemal ontoepaslik		

Dokumentering	<b>Interne dokumentering</b>				2				
	2		1				0		
	Kode regdeur program is voorsien van kommentaar/verklarende aantekeninge vir maklike interpretasie en verduideliking		Voorsien van kommentaar/verklarende aantekeninge, maar nie regdeur program nie				Geen kommentaar/verklarende aantekeninge nie		
Dokumentering	<b>Skeiding van afdelings</b>				2				
	2		1				0		
	Afdelings in kode van program duidelik geskei om leesbaarheid te verhoog (spasiëring, kommentaar vir metode/onderafdeling, ens.)		Sommige afdelings geskei				Nie geskei nie		
Algemene evaluering	<b>Tydbestuur</b>				3				
	3		2				1		0
	Alle spertye gehaal – al 3 fases en al die vereiste werk is gedoen		Haal 2 spertye of lewer betyds in, maar sommige werk is nie gedoen nie		Haal ten minste een sperdatum of betyds ingelewer, maar die meeste werk nie gedoen nie		Altyd laat, nooit gedoen nie		
	<b>Vermoë om kode wat lukraak gekies is, se doel en werking te verduidelik</b>				4				
	4		3				2		1
Verduidelik alle gekose kode duidelik en met selfvertroue; Toon uitstekende insig		Verduidelik gekose kode met geringe tekortkominge; Toon insig		Verduidelik sommige van die gekose kode; Toon 'n bietjie insig in sommige van die kode		Nie in staat om die meeste van die gekose kode toepaslik te verduidelik nie; Gebrek aan insig		Nie in staat om te verduidelik nie	
<b>Toepassing van stelsel in werklike situasie</b>				3					
3		2				1		0	
Die oplossing is 'n werkende stelsel wat in 'n werklike situasie gebruik kan word.		Die oplossing is 'n stelsel wat in 'n werklike situasie gebruik kan word met 'n bietjie aanpassing.		Sommige dele kan in 'n werklike situasie gebruik word.		Totaal irrelevant; Sal nie in 'n werklike situasie werk nie			
<b>Totaal:</b>					<b>110</b>				