

Bouwsteen: "Eerst denken dan doen."

Studiejaar 2
Versie 1.0, 2012-2013

Stenden hogeschool
Education
Van Schaikweg 94
Po box 2080
8701 CB Emmen
tel: 0591 853100
www.stenden.com

© Stenden hogeschool

Voorwoord

Versiebeheer

Versie	Datum	Omschrijving
1.0	9-11-2012	Initieel document

Inhoudsopgave

Voorwoord	2
Versiebeheer	3
1. Introductie	5
1.1. Bouwsteen rationale	5
1.2 Competenties	5
1.3 Bouwsteen thema	6
1.4 Bouwsteen doelstellingen	6
1.5 Voorkennis	7
1.6 Conventies	7
2 Toetsing	8
2.1 Algemeen	8
2.2 Uitleg studenten beoordeling	8
2.3 Actieve participatie	8
2.4 Bouwsteen herkansing	8
4 Structuur & Organisatie	10
4.1 Bouwsteen contact uren	10
Bijlage 1 Scoring rubrics Plan van aanpak/Software requirements specification / Testplan	12
Bijlage 2 Scoring rubrics onderzoek	13
Bijlage 3 beoordelingsformulier projectgroep	19
Bijlage 4 Code conventies	20

1. Introductie

1.1. Bouwsteen rationale

Kenmerkende beroepssituatie(s):

Je werkt bij een pensioenaanbieder die er aan denkt een informatiesysteem te realiseren dat er voor moet zorgen dat de klantafhandelingen door de medewerkers in het callcenter efficiënter gaat. Jij bent de vertegenwoordiger van de informatica afdeling in het onderzoektraject.

Je hebt geluk, want de dag na de aftrap van het project staat er in een krant een column over het functioneren van een helpdesk. De columnist heeft een gevoelige snaar geraakt, er volgen meerdere artikelen over helpdesks. Dit is prachtige informatie voor jullie project. Je gaat onderzoeken waar de onvrede vandaan komt. Je onderzoeksvraag is er op gericht om zo een goed advies uit te kunnen brengen voor het informatiesysteem dat je gaat realiseren.

Uiteindelijk breng je een advies uit aan de opdrachtgever. Dit advies is onderbouwd, je hebt je niet beperkt tot de krantenartikelen die na de column zijn verschenen maar hebt ook andere bronnen geraadpleegd, zo heb je zelf verschillende callcenter gebeld om als klant te ervaren hoe je wordt behandeld uiteraard heb je ook je eigen callcenter gebeld.

1.2 Competenties

Binnen deze bouwsteen wordt als beginnend beroepsprofessional gewerkt aan een aantal competenties. Omdat taken opgedeeld zijn, kan het voorkomen dat een competentie niet aan bod komt tijdens het project. Voor deze competentie geldt echter wel dat deze aan bod is geweest in een ondersteunend vak.

1.3 Bouwsteen thema

1.4 Bouwsteen doelstellingen

Binnen deze bouwsteen wordt gewerkt aan de volgende doelstellingen:

Projectmanagement:

1. De student kan een project organiseren.
2. De student kan een plan van aanpak maken.
3. De student kan een reële planning maken.

PGO/PBL:

4. De student is in staat om op een actieve en effectieve manier te communiceren met groepsleden door actief te luisteren en zijn/haar mening te delen met teamleden.
5. De student is in staat feedback te geven/ontvangen waarmee de student/ander zijn/haar voordeel kan doen.
6. De student is in staat de rol van de voorzitter op zich te nemen en de taken die bij deze rol en de rol van deelnemer horen te oefenen.
7. De student herkent hoe de communicatie over en weer in een vergadering beïnvloed wordt door groepsprocessen en groepsrollen.
8. De student kan een informatievergarend interview op correcte wijze uitvoeren.
9. De student kan reflecteren op zijn eigen bijdrage tijdens dit project.

Product:

10. De student kan een gebruikersinterface analyseren en aan de hand van zelfgemaakte schetsen ontwerpen en deze daarna realiseren.
11. De student kan een requirement analyse uitvoeren voor een eenvoudig informatiesysteem, volgens een standaard methode.
12. De student kan met gebruik van een ontwerptool een volledig ontwerp opstellen voor een informatiesysteem dat bestaat uit meerdere subsystemen en aansluit op bestaande software.
13. De student kan een informatiesysteem, dat bestaat uit meerdere subsystemen, bouwen, testen en beschikbaar stellen. Daarbij aansluiten op bestaande software en gebruikmaken van een ontwikkelomgeving.
14. De student kan een eenvoudig embedded systeem realiseren.
15. De student kan door middel van een brainstormsessie een aantal (maximaal 3) creatieve/innovatieve voorstellen presenteren voor een eindproduct.
16. De student realiseert een testplan.
17. De student realiseert een eenvoudig software requirement specification.

1.5 Voorkennis

De student dient:

1. De bouwsteen "Inleiding programmeren in Java" te volgen of gevolgd te hebben.
2. De bouwsteen "Inleiding in C#" te volgen of gevolgd te hebben.
3. De bouwsteen "Inleiding computernetwerken" te volgen of gevolgd te hebben.

1.6 Conventies

Een paar conventies met betrekking tot lettertypen zijn opgesteld om onderscheid te kunnen maken tussen tekst en programmeercode.

Normale tekst	Verdana. (tekengrootte 10)
Programma code	Courier new. (tekengrootte 10)
Programma output	Courier new. (tekengrootte 10)

Afspraken met betrekking tot naamgeving en indeling van programmacode kun je vinden in bijlage 4.

2 Toetsing

2.1 Algemeen

Om te beoordelen of je de doelstellingen van deze bouwsteen hebt behaald, krijgt de **groep** een cijfer voor het geleverde **eindproduct**. De docent gebruikt hierbij de scoring rubrics uit bijlage 3.

Aan de hand van dit cijfer moeten de studenten zelf onderling de individuele cijfers gaan bepalen (want zij weten precies hoe andere leden hebben gefunctioneerd in de groep) aan de hand van een aantal factoren en dit cijfer zal als advies dienen voor de docent. **Let wel, de docent is ten alle tijden verantwoordelijk voor het eindcijfer en hij/zij mag dan ook afwijken van het gegeven studentenadvies.**

Het is bekend dat studenten het lastig vinden om elkaar onderling te beoordelen. Toch is het belangrijk dat je dit kunt, omdat de kans groot is dat je daar ook mee in aanraking komt in de latere beroepspraktijk. De beoordeling mag je dan ook anoniem doen. Vanzelfsprekend worden onderdelen aangedragen waarop je moet letten tijdens de beoordeling.

2.2 Uitleg studenten beoordeling

Zoals aangegeven geeft de docent een cijfer voor het eindproduct. Nu moeten jullie aan de hand van dit cijfer alle groepsleden (inclusief jezelf) een cijfer geven. Hierbij moet je gebruik maken van het beoordelingshulpmiddel uit bijlage 4.

Voorbeeld:

De docent heeft bepaald dat het gemiddelde geleverde werk van de groep een 7 is.

Als er 4 groepsleden zijn dan krijg je een totaal van $4 \times 7 = 28$ punten. Deze 28 punten mag je per groepslid (minimaal 1, maximaal 10 en met een verschil van maximaal 2 van het docentcijfer) verdelen over alle groepsleden (dus inclusief jezelf). Je kunt nu bijvoorbeeld als cijfers geven: 6, 6.5, 7.5 en 8.

Vervolgens worden van alle groepsleden de gegeven cijfers opgeteld en worden deze gemiddeld. Wijkt een door een student gegeven cijfer meer af dan +/- 1,5 punt van het gemiddelde dan wordt de gehele beoordeling van deze student niet meegenomen en wordt het gemiddelde opnieuw berekend. Dit gebeurt net zo lang tot de afwijking binnen 1,5 punt ligt van het gemiddelde. (Mocht een student bewust zichzelf een te hoog cijfer geven of een andere student een te laag cijfer dan heeft dit geen invloed. Bijkomend verschijnsel is dat de bewuste student geen invloed meer heeft op het cijfer.

2.3 Actieve participatie

Voor deelname aan en beoordeling van deze bouwsteen is actieve participatie vereist. Zie de studiegids van jouw opleiding voor aanvullende regelgeving hieromtrent.

2.4 Bouwsteen herkansing

Studenten die bij de eerste beoordeling een onvoldoende gescoord hebben, kunnen vragen om een herkansing. Voor de actieve participatie geldt onder bepaalde voorwaarden een vervangende

opdracht, de opdracht zelf kan na beoordeling en feedback aangepast worden en voor een 2^e beoordeling worden aangeboden.

De herkansing moet binnen 1 week worden aangevraagd bij de docent welke verantwoordelijk is voor het project. De herkansingsopdracht(en) moet in de eerste week van **module periode 3** te worden ingeleverd/gepresenteerd.

4 Structuur & Organisatie

4.1 Bouwsteen contact uren

Het onderstaande schema geeft een overzicht van alle contacturen in deze bouwsteen.

Daarnaast wordt van studenten verwacht dat zij hun eigen (project)bijeenkomsten plannen waar zij kunnen werken aan de opdrachten. Dit geldt tevens voor de tijd die de student nodig heeft om (individuele) opdrachten voor te bereiden en te maken. Tevens geeft dit schema een goed overzicht van de verwachte studiebelasting per student.

Tabel 4.1: Student contact uren (SCU) per week:

Bouwsteen Project "Eerst denken dan doen: Student Contacturen en Student Belastinguren																					
Werkvorm	Aantal groepen	Week 1		Week 2		Week 3		Week 4		Week 5		Week 6		Week 7		Week 8		Week 9		Totaal	
		SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU	SCU	SBU
HC	6	1	3	1	3	1	3	1	3	1	3	1	3	1	3	1	3	0	0	8,00	24,00
WC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	-
ET	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	-
ZS	6	0	18	0	18	0	18	0	18	0	18	0	18	0	18	0	18	0	0	-	144,00
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	-
		1	21,00	1	21,00	1	21,00	1	21,00	1	21,00	1	21,00	1	21,00	1	21,00	-	-	8,00	168,00

HC = Hoorcollege
 WC = Werkcollege
 PBL = Problem based learning sessie
 ET = Eind toets
 ZS = Zelfstudie uren

SCU = Student Contact Uren (45 minuten)
 SBU = Studie Belasting Uren (60 minuten)

Bijlage 1 Scoring rubrics Plan van aanpak/Software requirements specification / Testplan

Onderdeel	1-3 punten	4-5 punten	6-7 punten	8-10 punten	Punten	Weeg-factor	Score
Plan van aanpak	Geen plan van aanpak. Niet gewerkt volgens het plan van aanpak	Plan van aanpak onvoldoende. Chaotisch en onduidelijk gewerkt binnen het project. Geen afstemming binnen het team. Geen terugkoppeling opdrachtgever/docenten.	De student stelt tijdens het project doelen (PVA), pakt werkzaamheden planmatig aan en reflecteert op het beroepsmatig handelen.	De student stelt tijdens het project realistische doelen (PVA), pakt werkzaamheden planmatig aan en reflecteert op het beroepsmatig handelen.		P*3	
Software Requirements Specification	Geen SRS.	SRS van onvoldoende kwaliteit. Chaotisch en onduidelijk.	SRS van voldoende kwaliteit.	SRS helder en duidelijk. Geeft een goed beeld van het te ontwikkelen product.		P*2	
Testplan	Geen testplan	Testplan van onvoldoende kwaliteit. Chaotisch en onduidelijk.	Testplan van voldoende kwaliteit.	Testplan geeft een duidelijk inzicht in hoe het eindproduct getest gaat worden. Testplan is volledig.		P*2	
Totaalscore							
Delen door						7	
Eindcijfer							

Bijlage 2 Scoring rubrics onderzoek

Scoring Rubrics						Score + toelichting
Vormgeving						
Titelblad	5	4	3	2	1	
	Alle onderdelen zijn aanwezig: <ul style="list-style-type: none">✓ Complete titel en evt. ondertitel✓ Naam onderzoeker(s)/auteur(s)✓ Datum en plaats✓ Instituut;✓ Naam coördinator en evt. vak	Er zijn 4 van de 5 onderdelen zijn aanwezig	Er zijn 3 van de 5 onderdelen zijn aanwezig	Er zijn 2 van de 5 onderdelen zijn aanwezig	Er is 1 van de 5 onderdelen aanwezig.	
Inhoudsopgave	De inhoudsopgave is: <ul style="list-style-type: none">✓ Aanwezig✓ Duidelijk✓ Geeft een overzichtelijk beeld van de opbouw van de tekst.✓ Is ordelijk vorm gegeven	De inhoudsopgave is aanwezig, duidelijk en geeft een overzichtelijk beeld van de opbouw van de tekst. De vormgeving is alleen niet ordelijk vormgegeven.	De inhoudsopgave is aanwezig en duidelijk, maar geeft geen overzichtelijk beeld van de tekst.	De inhoudsopgave is aanwezig, maar is rommelig en geeft geen overzichtelijk beeld van de opbouw van de tekst.	De inhoudsopgave is niet volledig. Er missen hoofdstukken/onderdelen.	
Literatuurlijst/ Bijlage	10	8	6	4	2	
	De literatuurlijst : <ul style="list-style-type: none">✓ Is geschreven volgens de APA-regels✓ Bevat alle verwijzingen die in de lopende tekst vernoemd worden De bijlage: <ul style="list-style-type: none">✓ Bevat aanvullende informatieve (tabellen, brieven, vragenlijsten, uitgebreide uitslagen)✓ Heeft een heldere nummering✓ Wordt duidelijk en volledig verwezen in de lopende tekst.	Er zijn 4 van de 5 onderdelen aanwezig.	Er zijn 3 van de 5 onderdelen aanwezig.	Er zijn 2 van de 5 onderdelen aanwezig.	Er is 1 van de 5 onderdelen aanwezig.	
Inhoud						
Samenvatting	10	8	6	4	2	

	De samenvatting bevat: <ul style="list-style-type: none"> ✓ De probleem- en doelstelling; ✓ Opzet, resultaten en belangrijkste conclusies; ✓ Relevante informatie En is kort en bondig geformuleerd.	De samenvatting bevat de probleem- en doelstelling, geeft de opzet, resultaten en belangrijkste conclusies correct weer en bevat alleen relevante informatie. De samenvatting is niet kort en bondig geformuleerd.	De samenvatting bevat de probleem- en doelstelling, geeft de opzet, resultaten en belangrijkste conclusies weer, maar bevat ook informatie die niet relevant is.	De samenvatting bevat de probleem- en doelstelling, maar de opzet, resultaten of belangrijkste conclusies worden niet voldoende uitgelicht.	De samenvatting bevat alleen de probleem – en doelstelling. De opzet, resultaten en belangrijkste conclusies worden niet uitgelicht.	
Inleiding	15	12	9	6	3	
	De inleiding beschrijft op een correcte manier de: <ul style="list-style-type: none"> ✓ De aanleiding tot onderzoek; ✓ Doel van het onderzoek; ✓ Probleemstelling en deelvragen; ✓ Leeswijzer ✓ Begripsafbakening; ✓ Theoretisch ondersteuning. 	De inleiding bevat de aanleiding, het doel, de probleemstelling en deelvragen, de leeswijzer en de begripsafbakening.	De inleiding bevat alle onderdelen behalve de begripsafbakening en de theoretische ondersteuning.	De inleiding bevat de aanleiding tot onderzoek, het doel van onderzoek en de probleemstelling en deelvragen.	De inleiding bevat alleen de aanleiding en het doel van onderzoek.	
Methode	10	8	6	4	2	
	De methode beschrijft: <ul style="list-style-type: none"> ✓ De populatie en steekproef ✓ Het meetinstrument ✓ De methode van dataverzameling ✓ Het verloop van evt. veldwerk ✓ Analysemethoden 	De methode bevat 4 van de 5 onderdelen.	De methode bevat 3 van de 5 onderdelen.	De methode bevat 2 van de 3 onderdelen.	De methode bevat 1 van de 5 onderdelen .	
Resultaten	15	12	9	6	3	
	De resultaten geven weer: <ul style="list-style-type: none"> ✓ Een beschrijving van de 	De resultaten geven een beschrijving van de onderzoeksgroep, geven de	De resultaten geven een beschrijving van de	Er is een beschrijving van de onderzoeksgroep	Er wordt alleen een beschrijving gegeven van de	

	onderzoeksgroep ✓ Kwantitatieve en/of kwalitatieve resultaten ✓ Tabellen, diagrammen en overzichten ✓ Een onafhankelijke weergave van de resultaten ✓ Géén conclusies	kwalitatieve en kwanitatieve resultaten weer, maken deze door tabellen en diagrammen duidelijk. En er worden géén conclusies gegeven.	onderzoeksgroep, geven de resultaten weer en tabellen en diagrammen geven een overzicht van de tekst.	en de resultaten worden weergegeven.	resultaten.	
Conclusie/discussie	10 De conclusie bevat: <ul style="list-style-type: none"> ✓ Een herhaling van de centrale vraag- en doelstelling ✓ Een koppeling van de resultaten naar de vraagstelling ✓ Een antwoord op de centrale vraagstelling ✓ Een evaluatie van methodologische resultaten ✓ Evt. aanbevelingen Eventueel een discussie over de uitkomsten.	8 De conclusie bevat een herhaling van de centrale vraag- en doelstelling, geeft een koppeling weer, beantwoordt de centrale vraagstelling en geeft een evaluatie van de methodologische resultaten.	6 De conclusie bevat een herhaling van de centrale vraag- en doelstelling, geeft een koppeling weer en beantwoordt de centrale vraagstelling.	4 De conclusie bevat een herhaling van de centrale vraag- en doelstelling en geeft een koppeling weer van de resultaten naar de vraagstelling	2 De conclusie bevat de centrale- en vraag en doelstelling.	
Algemene schrijfvaardigheden						
Algemene schrijfvaardigheden	20 Te tekst is: <ul style="list-style-type: none"> ✓ Helder, overzichtelijk een eenduidig 	16 De tekst bevat 4 van de 5 onderdelen	12 De tekst bevat 3 van de 5 onderdelen	8 De tekst bevat 2 van de 5 onderdelen	4 De tekst bevat 1 van de 5 onderdelen.	

	<ul style="list-style-type: none"> ✓ Aangepast op de doelgroep ✓ Gecontroleerd op spelling en grammatica ✓ Goed gestructureerd (paragraafindeling, alinea's) ✓ Op dezelfde manier vormgegeven ✓ 					
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--	--

Toelichting

Onderwerp	Onderdelen	Punten per onderdeel*	Totaal	Minimaal te halen punten**
Vormgeving	Titel Inhoudsopgave Literatuurlijst/bijlage	5 5 10	20	11
Inhoud	Samenvatting Inleiding Methode Resultaten Conclusie	10 15 10 15 10	60	33
Algemene schrijfvaardigheden	Algemene vaardigheden	20	20	11
Totaal		100	100	55

*Bij missende onderdelen of bij het missen van alle criteria worden 0 punten toegekend.

** Je moet op alle onderwerpen een voldoende halen. Dat betekent dat je 2/3 van de maximaal te behalen punten moet halen.

Bijlage 3 beoordelingsformulier projectgroep

Onderdeel	Punten	Weegfactor	Score
Eindcijfer Scoring rubrics Plan van aanpak/Software requirements specification / Testplan		1	
Eindcijfer onderzoek		1	
	Totaalscore		
	Delen door	2	
	Eindcijfer		

Bijlage 4 Code conventies

1. Naming

1.1 Use meaningful names.

Use descriptive names for all identifiers (names of classes, variables and methods. Avoid ambiguity. Avoid abbreviations.

Simple mutator methods should be named *setSomething(...)*.

Simple accessor methods should be named *getSomething(...)*.

Accessor methods with Boolean return values are often called *isSomething(...)*, for example *isEmpty()*.

1.2 Class names start with a capital letter.

1.3 Class names are singular nouns.

1.4 Methods and variable names start with lowercase letters.

All the – class, method and variable names – use capital letters in the middle to increase readability of compound identifiers, e.g. *numberOfItems*.

1.5 Constants are written in UPPERCASE.

Constants occasionally use underscores to indicate compound identifiers:

`MAXIMUM_SIZE`

2. Layout

2.1 One level of indentation is four spaces.

2.2 All statements within a block are intended one level.

2.3 Braces for classes and methods are alone on one line.

The braces for class and method blocks are on separate lines and are at the same indentation level, for example:

```
public int getAge()  
{  
    statements  
}
```

2.4 Also for all other blocks, are alone on one line.

The braces for all other blocks are on separate lines and are at the same indentation level, for example:

```
while(condition)  
{  
    statements  
}
```

```
if(condition)  
{  
    statements  
}  
else  
{  
    statements  
}
```

2.5 Always use braces in control structures

Braces are used in if-statements and loops even if the body is only a single statement.

2.6 Use a space before the opening brace of a control structure's block.

2.7 Use a space around operators.

2.8 Use a blank line between methods (and constructors).

Use blank lines to separate logical blocks of code. This means at least between methods, but also between logical parts within a method.