

Integration Test Plan  
versie 1.1.0



16 juni 2006

# Samenvatting

Dit is het Integration Test Plan (ITP) van het bICEPS project dat in het kader van het Software Engineering Project (2IP40) ontwikkeld wordt voor het bedrijf Sara Lee. Dit document bevat de procedures en informatie die nodig is om de integratie tests uit te voeren. Integratie testen verifiëren of de componenten op de juiste manier samenwerken zoals geformuleerd in het [ADD]. Het document voldoet aan de Software Engineering Standaard, zoals deze is opgesteld door de European Space Agency (ESA).

# Inhoudsopgave

<b>Samenvatting</b>	<b>II</b>
<b>Document status overzicht</b>	<b>IV</b>
<b>Document veranderingen overzicht</b>	<b>V</b>
<b>1 Introductie</b>	<b>1</b>
1.1 Doel	1
1.2 Scope	1
1.3 Lijst van definities	1
1.4 Referenties	2
<b>2 Test plan</b>	<b>3</b>
2.1 Test Items	3
2.2 Te testen eigenschappen	3
2.3 Test vereisten	3
2.4 Test taken	3
2.5 Omgevingseisen	4
2.6 Test case beoordelings criteria	4
<b>3 Test case specificaties</b>	<b>5</b>
3.1 ITC0 Integration Test	6
3.2 ITC1 Integration Test	8
3.3 ITC2 Integration Test	8
3.4 ITC3 Integration Test	9
3.5 ITC4 Integration Test	9
3.6 ITC5 Integration Test	10
<b>4 Test procedure</b>	<b>11</b>
4.1 ITP0 Integration Test Procedure	11
4.2 ITP1 Integration Test Procedure	11
4.3 ITP2 Integration Test Procedure	11
4.4 ITP3 Integration Test Procedure	11
4.5 ITP4 Integration Test Procedure	11
4.6 ITP5 Integration Test Procedure	12
<b>5 Test rapport</b>	<b>13</b>

# Document status overzicht

## Algemeen

Document titel: Integration Test Plan  
Identificatie: Documenten\ITP\ITP-1.1.0.pdf  
Auteur: Johan de Jong & Kevin Jonkers  
Document status: Intern goedgekeurd

## Document geschiedenis

Versie	Datum	Status
0.1.0	13-02-2006	Voorwaardelijk intern goedgekeurd
1.0.0	15-02-2006	Intern goedgekeurd
1.1.0	29-05-2006	Testresultaten toegevoegd

# Document veranderingen overzicht

## Algemeen

DCR number: 2  
Datum: 29-05-2006  
Auteur: Johan de Jong  
Goedgekeurd: Intern goedgekeurd  
Document titel: Integration Test Plan  
Identificatie: Documenten\ITP\ITP-1.1.0.pdf

## Veranderingen

Pagina	Paragraaf	Reden van verandering
13	Hoofdstuk 5	Testresultaten toegevoegd.

# Hoofdstuk 1

## Introductie

### 1.1 Doel

Dit document beschrijft het plan om te testen of de modules zoals gespecificeerd in het [ADD] op de juiste manier samenwerken. Het uitvoeren van deze testen zal in de ST fase plaats vinden.

### 1.2 Scope

Het systeem dat gemaakt gaat worden, gaat ICEPS heten, dit staat voor "International Coffee Expert Profile System". ICEPS is een systeem dat door koffie experts over de hele wereld gebruikt zal worden om de kwaliteit van de koffie te beoordelen. Het systeem zal door middel van een door de koffie-experts ingevulde vragenlijst een output naar excel en naar SAS generen, die door het bedrijf verder geanalyseerd kan worden. Daarnaast zal het ook mogelijk zijn om een grafische weergave van de resultaten te krijgen en deze te exporteren.

### 1.3 Lijst van definities

Afkorting	Betekenis
AD	Architectural Design
ADD	Architectural Design Document
DD	Detailed Design
SR	Software Requirements
SM	Senior Management
ST	System Transfer
ATP	Acceptance Test Plan
CI	Configuration Item
SL	Software Librarian
DDD	Detailed Design Document
ESA	European Space Agency
FTP	File Transfer Protocol
ITP	Integration Test Plan
PM	Project Manager
bICEPS	better International Coffee Expert Profile System
QE	Quality Engineer
URD	User Requirements Document
UTP	Unit Test Plan
VPM	Vice Project Manager
VQE	Vice Quality Engineer

## 1.4 Referenties

- [ESA] **ESA Software Engineering Standards** (ESA PSS-05-0 Issue 2),  
ESA Board for Software Standardization and Control (BSSC), 1991
- [SCMP] **Software Configuration Management Plan**,  
bICEPS,  
door Kevin Jonkers en Marcel Steine  
Technische Universiteit Eindhoven, 2005 / 2006
- [DDD] **Detailed Design Document**,  
bICEPS,  
door Stefan Rijkers, Arjan Kon  
Technische Universiteit Eindhoven, 2005 / 2006
- [SPMP] **Software Project Management Plan**,  
bICEPS,  
door Wouter van de Molengraft  
Technische Universiteit Eindhoven, 2005 / 2006
- [SQAP] **Software Quality Assurance Plan**,  
bICEPS,  
door Arjan Kon en Stefan Rijkers  
Technische Universiteit Eindhoven, 2005 / 2006
- [SRD] **Software Requirements Document**,  
bICEPS,  
door Kevin Verbeek, Mohammed El-Kebir, Hans Wauters, Johan de Jong  
Technische Universiteit Eindhoven, 2005 / 2006
- [ADD] **Architectural Design Document**,  
bICEPS,  
door Arjan Kon, Stefan Rijkers, Mohammed El-Kebir, Hans Wauter en Kevin  
Verbeek  
Technische Universiteit Eindhoven, 2005 / 2006

# Hoofdstuk 2

## Test plan

### 2.1 Test Items

De items die getest moeten worden, bestaan uit de integratie van de gecodeerde componenten die ontwikkeld zijn voor ICEPS. Voor het testen wordt gebruik gemaakt van een bottom-up aanpak. Het testen begint dus op het laagste niveau, waarna er steeds modules toegevoegd worden. De reden hiervoor is dat de low-level componenten vooral componenten zijn die data opslaan (zie [ADD], hoofdstuk 4.2). Wanneer je hiervoor stubs zou moeten schrijven die bruikbaar zijn om de tests goed uit te voeren zou je eigenlijk bijna geheel functionele componenten moeten schrijven hetgeen (onnodig) veel werk oplevert. De opbouw van ICEPS leent zich goed voor een bottom-up aanpak omdat de opbouw gelaagd is (geen circulaire referenties). Dit is gedaan omdat de ontwikkelingssoftware (Borland Delphi 7) een gelaagde structuur min of meer afdwingt. Het nadeel van de gekozen aanpak is dat het uiteindelijke systeem pas bestaat wanneer alle componenten zijn toegevoegd.

### 2.2 Te testen eigenschappen

ICEPS moet voldoen aan de requirements die opgesomd staan in [ADD, hoofdstuk 3].

### 2.3 Test vereisten

De volgende items moeten aangeleverd worden voor het testen begint:

- Het Software Verification and Validation Plan [SVVP].
- Het Architectural Design Document [ADD].
- Dit complete Integration Test Plan [ITP].
- Het Unit Test Plan [UTP].
- De modules die getest moet worden.

De volgende items moeten geleverd worden na het testen:

- Integration Test Report.
- Integration Test output data.
- Probleemrapporten (indien nodig).

### 2.4 Test taken

De volgende taken zijn noodzakelijk om de integratie tests uit te voeren:

- Ontwerpen integratie tests.
- Ontwerpen van drivers (voor zover dit nog niet gebeurd is bij de unit tests).

- Ontwerpen van input data (voor zover dit nog niet gebeurd is bij de unit tests).
- Verzekeren dat aan alle omgevingseisen voldaan is.
- De unit tests moeten zijn uitgevoerd.
- Het doen van de integration tests.
- Het schrijven van een testrapport en indien nodig een probleemrapport. probleemrapport.

## 2.5 Omgevingseisen

De omgevingseisen staan in [SRD, hoofdstuk 3].

## 2.6 Test case beoordelings criteria

Ieder test scenario moet beschrijven wat de criteria zijn om te slagen voor de specifieke test.

## Hoofdstuk 3

# Test case specificaties

De componentenopbouw van ICEPS bestaat uit een set componenten die weinig onderlinge verbindingen hebben en één component die alles omvat. Om die reden zullen bij de eerste integratie test case meteen drie componenten in één keer samengevoegd worden. Ook zullen de meeste integratietesten alleen maar uit regressie bestaan (controleren of het nieuwe component geen conflicten veroorzaakt bij de reeds toegevoegde componenten) omdat het nieuwe component geen verbindingen heeft met de andere. De laatste integratietest zal de grootste zijn, omdat het component 'ICEPS' verbonden is met alle andere componenten. Er zal regelmatig verwezen worden naar het bestaande systeem. Hiermee wordt bedoeld: het geheel van componenten die reeds geïntegreerd zijn.

### 3.1 ITC0 Integration Test

<b>Test Case Identificatie</b>	ITC0
<b>Test Items</b>	Samenvoegen van de componenten 'Attributes', 'LabData' en 'Session'.
<b>Input Specification</b>	<p>De volgende handelingen moeten worden uitgevoerd voor protected attribuutlijsten, custom attribuutlijsten en lab attribuutlijsten:</p> <ul style="list-style-type: none"> <li>• Attribuut toevoegen aan de lijst</li> <li>• Attribuut uitschakelen in de lijst</li> <li>• Attribuut opvragen op naam</li> <li>• Attribuut opvragen op index</li> <li>• Attribuut wissen van de lijst</li> <li>• Commit naar database voor al het bovenstaande</li> </ul> <p>De volgende handelingen moeten uitgevoerd worden met protected attribuutlijsten, custom attribuutlijsten en lab attribuutlijsten:</p> <ul style="list-style-type: none"> <li>• Toevoegen lijst aan de database</li> <li>• Verwijderen lijst uit de database</li> </ul> <p>De volgende handelingen moeten uitgevoerd worden op protected attributen, custom attributen en lab attributen:</p> <ul style="list-style-type: none"> <li>• Naam attribuut wijzigen</li> <li>• Niet-beschikbaar maken van het attribuut</li> <li>• Commit naar database voor al het bovenstaande</li> </ul> <p>De volgende handelingen moeten uitgevoerd worden op een sample:</p> <ul style="list-style-type: none"> <li>• Opmerkingen veranderen</li> <li>• Scores veranderen</li> <li>• Labscores veranderen</li> <li>• Eindcijfer veranderen</li> <li>• Sample als 'bad sample' markeren</li> <li>• Commit naar database voor al het bovenstaande</li> </ul>

	<p>De volgende handelingen moeten uitgevoerd worden voor een sessie:</p> <ul style="list-style-type: none"> <li>• Sessie toevoegen aan de database</li> <li>• Sessie wissen van de database</li> <li>• Sessie opvragen op naam</li> <li>• Sessie opvragen op index</li> </ul> <p>De volgende handelingen moeten uitgevoerd worden op een sessie:</p> <ul style="list-style-type: none"> <li>• Beschrijving veranderen</li> <li>• Aanmaakdatum invullen</li> <li>• Sample toevoegen</li> <li>• Sample dupliceren</li> <li>• Sample wissen</li> <li>• Sessie finalizen</li> <li>• Sessie exporteren naar Excel</li> <li>• Overview van de sessie afdrukken</li> <li>• Commit naar database</li> </ul>
<b>Output Specification</b>	<p>Bij het uitvoeren van ieder van de bovenstaande handelingen moet de database de uitgevoerde veranderingen reflecteren en steeds consistent blijven. Hierbij dient opgemerkt te worden dat handelingen alleen zichtbaar zijn in de database op de plaatsen waarbij expliciet genoemd staat dat er interactie is met de database ; andere operaties zijn indirect zichtbaar doordat er eerst een commit uitgevoerd moet worden of vragen alleen informatie op. Deze laatste dienen het juiste resultaat op te leveren.</p>
<b>Controleer</b>	<p>Database consistent en juist bijgewerkt na iedere veranderingsoperatie en resultaten consistent na iedere opvraagoperatie.</p>

### 3.2 ITC1 Integration Test

<b>Test Case Identificatie</b>	ITC1
<b>Test Items</b>	Samenvoegen van het bestaande systeem en component 'UserReg'. Dit component heeft geen verbindingen met het bestaande systeem, anders dan dat zowel het bestaande systeem als 'UserReg' gebruik maken van de database (ADO). Dit is dus vooral een regressie-test.
<b>Input Specification</b>	<p>De volgende handelingen moeten uitgevoerd worden:</p> <ul style="list-style-type: none"> <li>• Nieuwe gebruiker registreren</li> <li>• Wachtwoord van een gebruiker veranderen</li> <li>• Gebruiker opvragen op index</li> <li>• Gebruiker opvragen op gebruikersnummer</li> </ul> <p>Naast deze tests moeten ook regressietests worden uitgevoerd (alle tests van de vorige test case moeten opnieuw worden uitgevoerd om te controleren of het toevoegen van dit component geen nieuwe problemen heeft geïntroduceerd in de andere componenten).</p>
<b>Output Specification</b>	Bij de veranderingsoperaties moet de database consistent blijven en het gevraagde gewijzigd zijn. Bij het opvragen van gegevens moeten de gegevens kloppen met datgene wat al in de database staat.
<b>Controleer</b>	Database consistent na iedere wijzigingsoperatie en resultaten consistent na iedere opvraagoperatie.

### 3.3 ITC2 Integration Test

<b>Test Case Identificatie</b>	ITC2
<b>Test Items</b>	Samenvoegen van het bestaande systeem en de component 'Logs'. Dit component heeft geen verbindingen met het bestaande systeem, anders dan dat zowel het bestaande systeem als 'Logs' gebruik maken van de database (ADO). Dit is dus vooral een regressie-test.
<b>Input Specification</b>	<p>De volgende handelingen moeten uitgevoerd worden:</p> <ul style="list-style-type: none"> <li>• Toevoegen van een log message</li> <li>• Opvragen van een log message</li> </ul> <p>Naast deze tests moeten ook regressietests worden uitgevoerd (alle tests van de vorige test case moeten opnieuw worden uitgevoerd om te controleren of het toevoegen van dit component geen nieuwe problemen heeft geïntroduceerd in de andere componenten).</p>
<b>Output Specification</b>	Bij het opvragen van de log messages moeten de messages die erin staan kloppen met de log messages die zijn toegevoegd. Ook de volgorde moet hetzelfde zijn.
<b>Controleer</b>	De database moet alle berichten die toegevoegd zijn in dezelfde volgorde bevatten. Bij het opvragen van berichten moeten de berichten in de juiste volgorde uit de klasse TLoggerAdmin komen (zoals ze in de database staan).

### 3.4 ITC3 Integration Test

<b>Test Case Identificatie</b>	ITC3
<b>Test Items</b>	Samenvoegen van het bestaande systeem en de component 'Language'. Dit component heeft geen verbindingen met het bestaande systeem, anders dan dat zowel het bestaande systeem als 'Language' gebruik maken van de database (ADO). Dit is dus vooral een regressie-test.
<b>Input Specification</b>	Het inlezen van een language pack moet uitgevoerd worden. Naast deze tests moet ook regressietests worden uitgevoerd (alle tests van de vorige test case moeten opnieuw worden uitgevoerd om te controleren of het toevoegen van dit component geen nieuwe problemen heeft geïntroduceerd in de andere componenten).
<b>Output Specification</b>	Het language pack moet ingelezen zijn en de user interface moet naar de ingelezen taal veranderd zijn.
<b>Controleer</b>	Het language pack moet ingelezen zijn. Aangezien de user interface op dit moment nog niet aangesloten is, kan nog niet gezien worden dat de user interface van taal verandert.

### 3.5 ITC4 Integration Test

<b>Test Case Identificatie</b>	ITC4
<b>Test Items</b>	Samenvoegen van het bestaande systeem en de componenten 'Visualisation' en 'PCAPackage'. Het component 'Visualisation' maakt gebruik van de componenten 'Session' en 'PCAPackage'. De samenwerking tussen deze twee zal getest worden in dit component, samen met regressietesten om te verifiëren dat dit component geen problemen introduceert in het bestaande systeem.
<b>Input Specification</b>	De volgende handelingen moeten uitgevoerd worden: <ul style="list-style-type: none"> <li>• Genereren van een grafiek</li> <li>• Genereren van een PCA-mapping</li> <li>• Opvragen van een grafiek</li> <li>• Opvragen van een PCA-mapping</li> <li>• Verwijderen van een grafiek</li> <li>• Verwijderen van een PCA-mapping</li> </ul>

	<p>De volgende handelingen moeten op een PCA-mapping uitgevoerd worden:</p> <ul style="list-style-type: none"> <li>• Sample eigenschappen veranderen (kleur, aangevinkt)</li> <li>• Sample eigenschappen opvragen (kleur, aangevinkt)</li> <li>• Opnieuw berekenen van de PCA-mapping</li> <li>• Afdrukken van de PCA-mapping</li> <li>• Exporteren van de PCA-mapping naar BMP of JPG</li> <li>• Assen verwisselen</li> <li>• Pannen</li> <li>• Zoomen</li> </ul> <p>De volgende handelingen moeten op een grafiek uitgevoerd worden:</p> <ul style="list-style-type: none"> <li>• Sample eigenschappen veranderen (kleur, aangevinkt)</li> <li>• Sample eigenschappen opvragen (kleur, aangevinkt)</li> <li>• Opnieuw berekenen van de grafiek</li> <li>• Afdrukken van de grafiek</li> <li>• Exporteren van de grafiek naar BMP of JPG</li> </ul>
<b>Output Specification</b>	De uitvoer zal bestaan uit PCA-mappings, grafieken, BMP-bestanden en JPG-bestanden. Andere uitvoer is de eigenschappen van een sample geassocieerd met een grafiek of PCA-mapping bij het opvragen van de eigenschappen hiervan (kleur, aangevinkt).
<b>Controleer</b>	De uitvoer moet overeenkomen met datgene wat in de database staat. Bij het veranderen van sample eigenschappen geassocieerd met een grafiek of PCA-mapping moet de database deze verandering reflecteren en de database moet uiteraard consistent blijven.

### 3.6 ITC5 Integration Test

<b>Test Case Identificatie</b>	ITC5
<b>Test Items</b>	Samenvoegen van het bestaande systeem en de component 'ICEPS'. Dit is de grootste integratie, omdat het component 'ICEPS' verbonden is met alle andere componenten. Pas na het toevoegen van 'ICEPS' bestaat het systeem echt en kan het met de user interface aangestuurd worden.
<b>Input Specification</b>	Alle voorgaande tests moeten opnieuw uitgevoerd worden, alleen nu vanuit de grafische user interfaces in plaats van met drivers.
<b>Output Specification</b>	De uitvoer is hetzelfde als bij de vorige tests, behalve dat de uitvoer nu gedeeltelijk gegeven wordt door de grafische user interface.
<b>Controleer</b>	De controles zijn hetzelfde als bij de voorgaande tests, behalve dat nu alles ook in de user interface gecontroleerd moet worden. Het controleren van de database moet uiteraard ook weer gebeuren.

# Hoofdstuk 4

## Test procedure

Bij het integreren van de componenten worden bij de eerst test case de eerste drie componenten geïntegreerd. Bij de daaropvolgende test cases wordt steeds één component toegevoegd totdat het systeem compleet is. Om die reden is het van belang dat de test cases in dezelfde volgorde uitgevoerd worden als dat ze in het vorige hoofdstuk en dit hoofdstuk genoemd staan.

Er zal regelmatig verwezen worden naar het bestaande systeem. Hiermee wordt bedoeld: het geheel van componenten die reeds geïntegreerd zijn.

### 4.1 ITP0 Integration Test Procedure

<b>Test Case Identificatie</b>	ITP0
<b>Doel</b>	Deze test procedure test de communicatie tussen de componenten 'Attributes', 'LabData' en 'Session'.
<b>Procedure stappen</b>	Voer uit: ITC0

### 4.2 ITP1 Integration Test Procedure

<b>Test Case Identificatie</b>	ITP1
<b>Doel</b>	Deze test procedure test de samenwerking tussen het bestaande systeem en het component 'UserReg'.
<b>Procedure stappen</b>	Voer uit: ITC1

### 4.3 ITP2 Integration Test Procedure

<b>Test Case Identificatie</b>	ITP2
<b>Doel</b>	Deze test procedure test de samenwerking tussen het bestaande systeem en het component 'Logs'.
<b>Procedure stappen</b>	Voer uit: ITC2

### 4.4 ITP3 Integration Test Procedure

<b>Test Case Identificatie</b>	ITP3
<b>Doel</b>	Deze test procedure test de samenwerking tussen het bestaande systeem en het component 'Language'.
<b>Procedure stappen</b>	Voer uit: ITC3

### 4.5 ITP4 Integration Test Procedure

<b>Test Case Identificatie</b>	ITP4
<b>Doel</b>	Deze test procedure test de communicatie tussen de componenten 'Session', 'Visualisation' en 'PCAMapping'.
<b>Procedure stappen</b>	Voer uit: ITC4

## 4.6 ITP5 Integration Test Procedure

<b>Test Case Identificatie</b>	ITP5
<b>Doel</b>	Deze test procedure test de communicatie tussen de component 'ICEPS' en de rest van de componenten.
<b>Procedure stappen</b>	Voer uit: ITC5

## Hoofdstuk 5

# Test rapport

<b>Test Case Indentificatie</b>	ITP0
<b>Test Resultaat</b>	Geslaagd
<b>Opmerkingen</b>	-
<b>Test Case Indentificatie</b>	ITP1
<b>Test Resultaat</b>	Geslaagd
<b>Opmerkingen</b>	-
<b>Test Case Indentificatie</b>	ITP2
<b>Test Resultaat</b>	Geslaagd
<b>Opmerkingen</b>	-
<b>Test Case Indentificatie</b>	ITP3
<b>Test Resultaat</b>	Niet geslaagd
<b>Opmerkingen</b>	Deze component hoeft niet geïmplementeerd te worden.
<b>Test Case Indentificatie</b>	ITP4
<b>Test Resultaat</b>	Geslaagd
<b>Opmerkingen</b>	-
<b>Test Case Indentificatie</b>	ITP5
<b>Test Resultaat</b>	Geslaagd
<b>Opmerkingen</b>	-