



Online-Library	Version: 1.0
Functions Points	Date: 8 April 2014

# Inhaltsverzeichnis

1.	Einführung	3
1.1	Bestimmung	3
1.2	Einschränkungen	3
2.	Berechnung	3
2.1	Use-Case Login	3
2.2	TinyTools	4
2.3	Diagramme	6
3.	Abschätzung: Use-Case “Eingeben”	7
4.	Abschätzung: Use-Case „Suchen“	7

Online-Library	Version: 1.0
Functions Points	Date: 8 April 2014

# Function Points

## 1. Einführung

Function Points sind eine Größe, die die Komplexität von Software repräsentiert. Sie sind unabhängig von der Programmiersprache und fassen viele Faktoren, wie z.B. Eingänge und Ausgänge in einer Zahl zusammen. Mit ihrer Hilfe können aus bereits implementierten Projekten bzw. Use Cases Abschätzungen von Kenngrößen wie Aufwand und Lines of Code (LOC) für noch folgende Projekte bzw. Use Cases erstellt werden.

### 1.1 Bestimmung

Zur Bestimmung werden die Anzahl der Internal Logical Files (ILF), External Interface Files (EIF) External Inputs (EI), External Output (EO) und External Inquiries (EI) gewichtet miteinander verrechnet. Die Gewichtung wird über die Komplexität festgelegt

### 1.2 Einschränkungen

Function Points wurden für Programme mit Datenbankbindung konzipiert. Programme mit komplexer Kommunikation zwischen mehreren Teilnehmern, aber ohne Datenbank erhalten nur wenige Funktion Points, obwohl sie trotzdem sehr komplex sind. Deswegen haben wir bei unseren Use Cases die Komplexität der Files nicht nach der herkömmlichen Methode berechnet, sondern nach der Anzahl der Teilnehmer. Dabei wurden Files die nur den Server betreffen mit Low eingestuft und welche die Server und einen Client betreffen mit Average. Bei Files die den Server und mehrere Clients betreffen wurde die Komplexität High gewählt. Mit dieser Methode erzielten wir erheblich bessere Ergebnisse.

## 2. Berechnung

### 2.1 Use-Case Login

<b><u>Communication</u></b>	
Complexity	Average (Server+Client)
<b><u>Files</u></b>	
<b>ILF1</b>	ApplicationUser
DETs	5
Complexity	Low (5)
<b>ILF2</b>	RegisterViewModel
DETs	7
Complexity	Low (7)
<b><u>External Input (EI)</u></b>	
<b>EI1</b>	Registrieren-Formular
File-Type-References	1 – Identity Model
DETs	8
Complexity	Low (8)
<b><u>External Output (EO)</u></b>	
<b>EO1</b>	Startseite
File Type References	0
DETs	0
Complexity	Low (0)

- ➔ Function Points: 19,32
- ➔ LOC: 147

Online-Library	Version: 1.0
Functions Points	Date: 8 April 2014

## 2.2 TinyTools

### Domain Characteristic Table

MEASUREMENT PARAMETER	COUNT (value >= 0)	WEIGHTING FACTOR		
		Simple	Average	Complex
Number of User Input	<input type="text" value="1"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Number of User Outputs	<input type="text" value="1"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Number of User Inquiries	<input type="text" value="0"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Number of Files	<input type="text" value="2"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Number of External Interfaces	<input type="text" value="0"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

[Complexity Adjustment Table](#) | [FP Calculation](#)

Online-Library	Version: 1.0
Functions Points	Date: 8 April 2014

## Complexity Adjustment Table

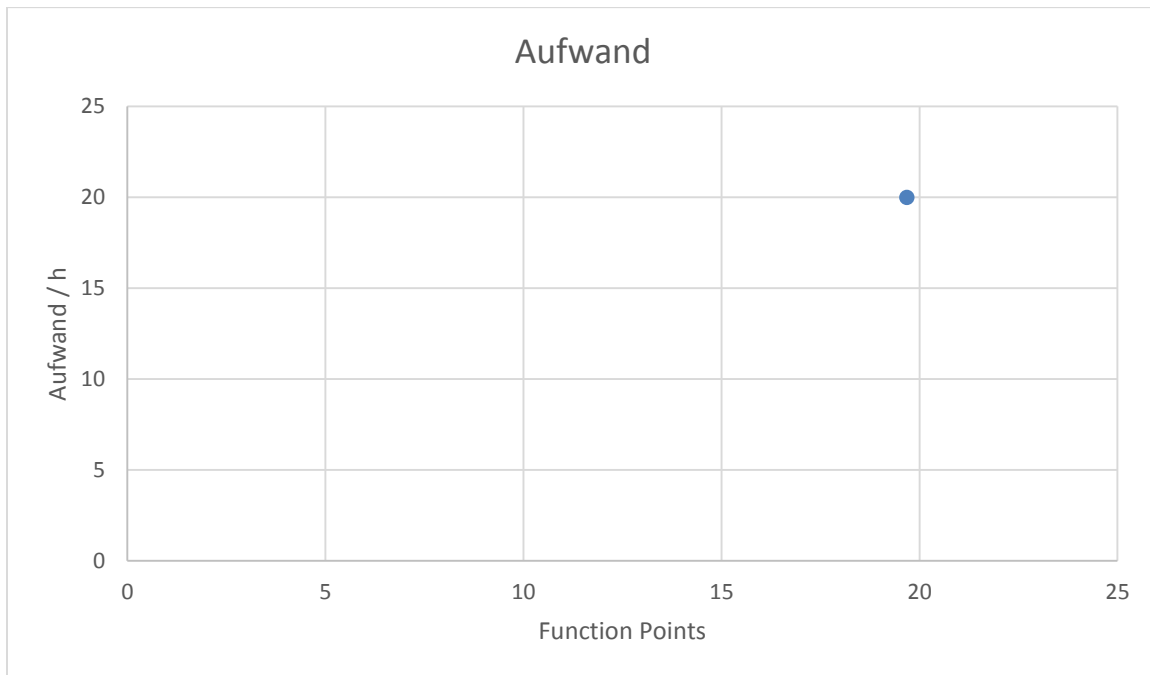
ITEM	COMPLEXITY ADJUSTMENT QUESTIONS	SCALE					
		No Influence					Essential
		0	1	2	3	4	5
1	Does the system require reliable backup and recovery?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2	Are data communications required?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
3	Are there distributed processing functions?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4	Is performance critical?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5	Will the system run in an existing, heavily utilized operational environment?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
6	Does the system require on-line data entry?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7	Does the on-line data entry require the input transaction to be built over multiple screens or operations?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8	Are the master files updated on-line?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
9	Are the inputs, outputs, files or inquiries complex?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
10	Is the internal processing complex?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11	Is the code to be designed reusable?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
12	Are conversion and installation included in the design?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
13	Is the system designed for multiple installations in different organizations?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
14	Is the application designed to facilitate change and ease of use by the user?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>

[Domain Characteristic Table](#) | [FP Calculation](#)

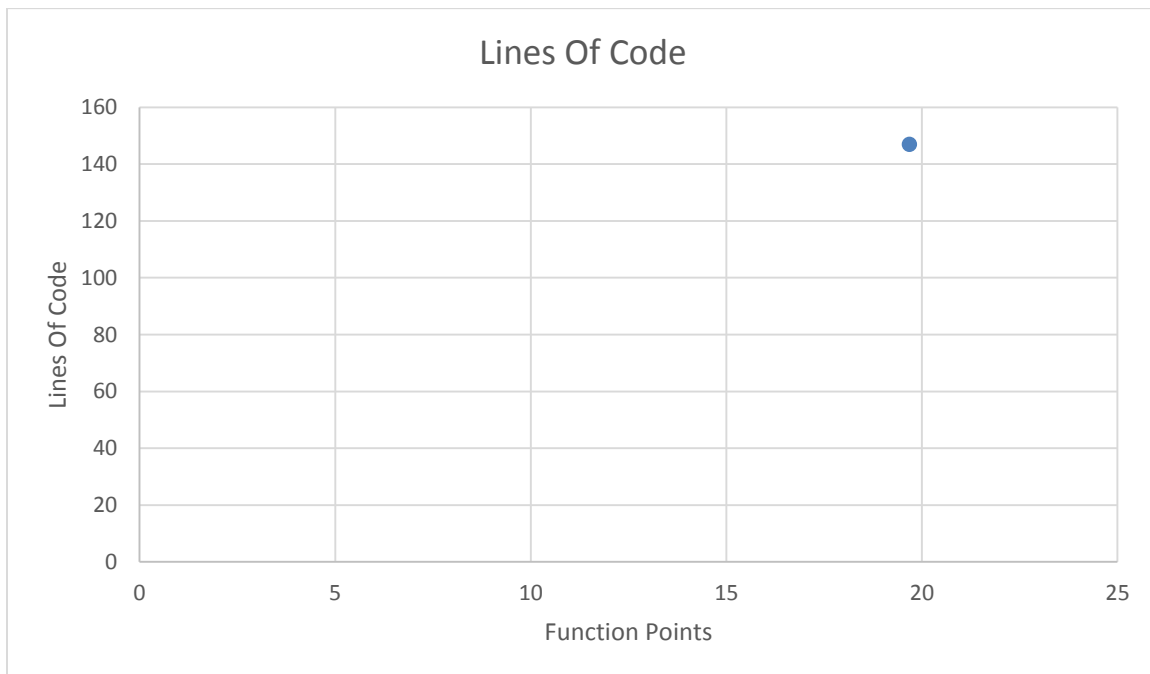
RESULT	
PROJECT FUNCTION POINTS	19.32

[Top of Page](#) | [Domain Characteristic Table](#) | [Complexity Adjustment Table](#)

## 2.3 Diagramme



**Abbildung 1: Aufwand**



**Abbildung 2: Lines Of Code**

Online-Library	Version: 1.0
Functions Points	Date: 8 April 2014

### 3. Abschätzung: Use-Case “Eingeben”

Der Aufwand kann anhand der Analyse der Registrierung abgeschätzt werden. Da für die Registrierung viel Code automatisch generiert wurde, wird der Aufwand beim Eingeben vermutlich höher sein, während die Function-Points ebenfalls auf etwa 19-20 geschätzt werden können. Die Lines of Code können auf ca. 140 geschätzt werden.

### 4. Abschätzung: Use-Case „Suchen“

Da die Suche nach einzelnen Titeln ein relativ einfacher Use-Case ist liegen die Function Points geschätzt bei 12-15. Auch die Lines of Code liegen vermutlich deutlich unter den 140 aus Abbildung 2. Hier wird ein Wert um die 100 erwartet.