

# Project Report - TravelDream

Paolo Manca - Francesco Nero

February 14, 2014

## Contents

<b>1</b>	<b>Function Points</b>	<b>2</b>
1.1	External logic files . . . . .	2
1.2	Internal logic files . . . . .	2
1.3	Inquiries . . . . .	3
1.4	External inputs . . . . .	4
1.5	External outputs . . . . .	4
1.6	Considerations . . . . .	4
<b>2</b>	<b>COCOMO</b>	<b>5</b>
2.1	Estimates . . . . .	5
2.2	Actual . . . . .	5
2.3	Considerations . . . . .	5
<b>3</b>	<b>Conclusions</b>	<b>6</b>

# 1 Function Points

## 1.1 External logic files

No external application are used currently, so the total function points from external logic files is 0.

## 1.2 Internal logic files

Table 1: Internal logic files

Entity	Type	FP
FinalExcursion	Simple	7
FinalFlight	Simple	7
FinalHotel	Simple	7
FinalPackage	Medium	10
Package	Medium	10
Product	Simple	7
User	Medium	10
IdPool	Medium	10
TOT		68

### 1.3 Inquiries

Table 2: Inquiries

<b>Inquiry</b>	<b>Actor</b>	<b>Type</b>	<b>FP</b>
View all final packages	EMPLOYEE	Medium	4
View final package	EMPLOYEE	Medium	4
View my final package	CUSTOMER	Medium	4
View shared package	EVERYONE	Medium	4
View final package by user	CUSTOMER/EMPLOYEE	Medium	4
View final product	CUSTOMER	Simple	3
View package	CUSTOMER/EMPLOYEE	Medium	4
View all packages	CUSTOMER/EMPLOYEE	Medium	4
View product	CUSTOMER/EMPLOYEE	Simple	3
View all products	CUSTOMER/EMPLOYEE	Simple	3
Search product	CUSTOMER/EMPLOYEE	Medium	4
View user	EMPLOYEE	Simple	3
TOT			44

## 1.4 External inputs

Table 3: External inputs

Input	Actor	Type	FP
Login/Logout	CUSTOMER/EMPLOYEE	Simple	3
Register	CUSTOMER	Medium	4
Create product	EMPLOYEE	Simple	3
Create package	EMPLOYEE	Medium	4
Register employee	EMPLOYEE	Simple	3
Remove user	EMPLOYEE/CUSTOMER	Medium	4
Modify account information	EMPLOYEE/CUSTOMER	Simple	3
Modify package	EMPLOYEE	Medium	4
Finalize Package	CUSTOMER	Medium	4
Add/Remove product to package	EMPLOYEE	Medium	4
Reserve final package	CUSTOMER	Simple	3
Share final package	CUSTOMER	Medium	4
Copy shared package	CUSTOMER	Complex	6
Add/Remove product to final package	CUSTOMER	Medium	4
Finalize product	CUSTOMER	Medium	4
Remove final product from final package	CUSTOMER	Simple	3
Modify final product	CUSTOMER	Simple	3
Modify product	EMPLOYEE	Simple	3
TOT			66

## 1.5 External outputs

No particular complex objects are produced by the system, at the current stage of implementation.

## 1.6 Considerations

The total number of function points is then:  $68 + 44 + 66 = 178$ . The estimated KLOC are then (without applying corrections) subdivided between XHTML/JSF (language factor 9) and JAVAEE (language factor 46) code, assuming a ratio of 2:3.

$$LOC = 5553.6 = 9 * 178 * 0.4 + 46 * 178 * 0.6$$

The value for the XHTML/JSF language factor was calculated starting from the known HTML language factor, and dividing it by a factor (3.7) which we empirically measured from

our application (number of html LOC produced by the corresponding JSF code over the LOC of the JSF page).

Table 4: HTML Language factor

Language	Avg	Median	Low	High
HTML	34	40	14	48

## 2 COCOMO

### 2.1 Estimates

We calculate here the COCOMO estimations values, using the basic model on an organic type of project.

$$EFFORT = a_b S^{b_b} = 2.4 * 5.5536^{1.05} = 14.5216 [person/month]$$

$$DURATION = c_b S^{d_b} = 2.5 * 5.5536^{0.38} = 4.7960 [months]$$

$$STAFF = EFFORT/DURATION = 3.0279 [persons]$$

### 2.2 Actual

We report here the real values recorded for a comparison (We considered an 8 hour work day and a 40 hour work week to calculate the DURATION).

$$DURATION = 1 [months]$$

$$STAFF = 2 [persons]$$

$$EFFORT = 2 [person/month]$$

### 2.3 Considerations

There is obviously a big difference between COCOMO estimates and the actual recorded values. We think these could be the main reasons for this divergence:

- the actual number of lines of code (which is very close to the FP estimate) is inflated by a lot of auto-generated code by the IDE
- the basic COCOMO model is not very precise

- most of the development was focused on implementing features (typical for the start of a project), and not maintaining them, thus a greater number of added LOC
- the values used for COCOMO estimation were guessed (the project might be simpler than organic), so they probably introduce a lot of error
- the values used for FP calculation are also inferred

### 3 Conclusions

Looking at the current state of the project and taking in consideration the experience acquired during these months, we can point out some weaknesses in the taken approach:

- too many features, at the expense of testing and documentation (e.g. lack of a proper use manual), were implemented for the delivery. A more effective approach would have been to postpone most of the non-essential functions to a future release, and to do proper testing and documentation on the core features.
- creating JPA entities from the database schema (which was laid out in the design phase) made some of java's strengths useless (flattened object hierarchies). We started the refactoring process (to reach a more object oriented design) halfway through the implementation, which caused a delay in all of the other planned development tasks. The refactoring is still in process.