

## Homework 5

Name : Kevin Dsa

CWID: 20009000

In this assignment, you need to choose one of the following two options and perform a function point analysis to derive effort estimates for the project selected:

(Option 1) Any ongoing software development projects that you are currently involved in. It may be your course project from another course. Please provide a brief description of the project selected.

(Option 2) If you don't have any, you can use this case project stated in the following. You are to design and estimate a system for reserving theater tickets. It has a database for tickets, a system admin interface, a user interface, and an interface to an on-line credit card validation system. The credit-card validation system is a different system that you do not need to build.

Your tasks are:

A. Create a high-level block diagram design for this system.

B. Conduct IFPUG function point analysis, including:

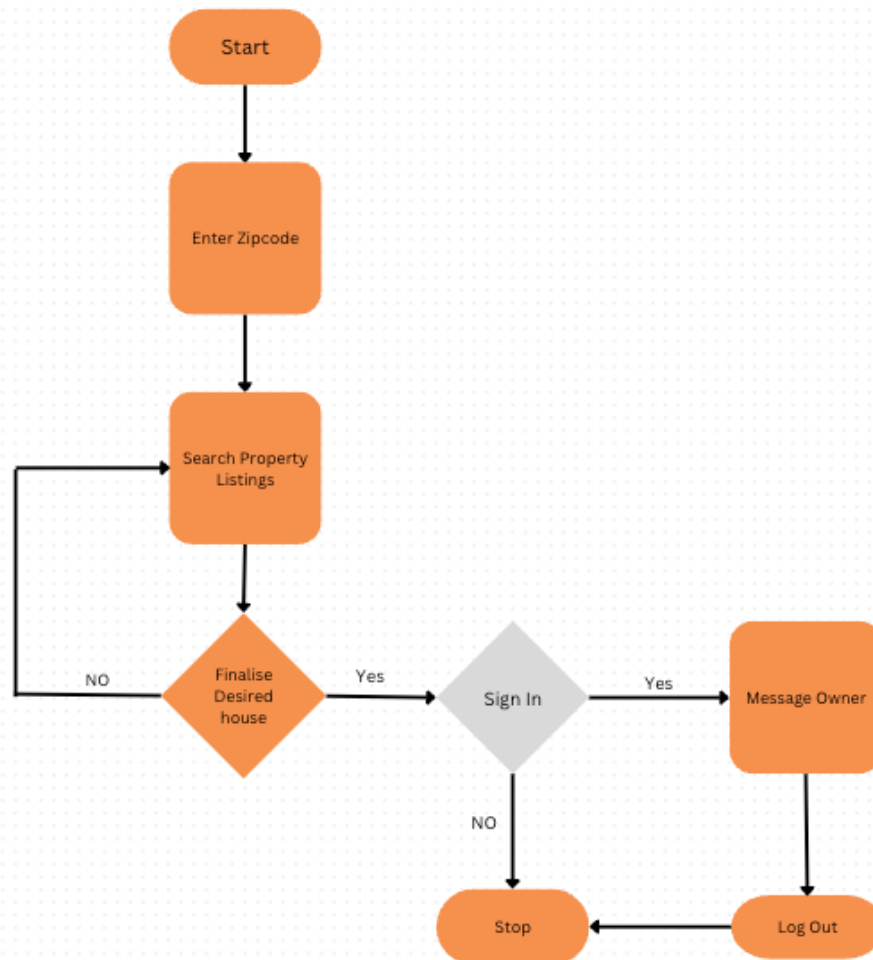
1. Count the number of function points.
2. How much effort do you estimate, based upon this estimate (do not convert to LOC yet)?
3. Assume Java is the language. How many LOC do you estimate?
4. Using Jones' table, what is the estimated effort?
5. Your boss listens nicely to your estimates and tells you they are 25% too large. How do you cut the effort by 25%? Hint: Change your design/plan so that there really is 25% less effort rather than working overtime or just saying a component is simpler. Think about the factors that are used specifically in the function point algorithm and the LOC algorithm. How do you reduce the expected LOC without cutting functionality?

### Option 1:

Project Name: RentPipe

Project Description: Rent Pipe is an application that allows users to find properties that are listed for Sale or Rent in their desired locations. This application has a Database system to store Images and description of the houses. It also has a Messaging feature where the user is able to get in contact with the owner of the house.

A. Create a high-level block diagram design for this system.



B. Conduct IFPUG function point analysis

1. Count the number of function points.

Used AFP process for Weight(From Slides)

Function Point Calculator				Total	Factor	FP
The Madison Utilities, Department of Computer Science, James Madison University				549	0.81	445

Direct Measure	Simple	Count		Weighted Measure	Value Adjustment Factor	0	1	2	3	4	5
		Average	Complex								
External Inputs (EIs)	<input type="text" value="0"/>	<input type="text" value="35"/>	<input type="text" value="0"/>	140	The system requires reliable backup and recovery.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
External Outputs (EOs)	<input type="text" value="0"/>	<input type="text" value="40"/>	<input type="text" value="0"/>	200	Specialized data communications are required.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
External Inquiries (EQs)	<input type="text" value="0"/>	<input type="text" value="1"/>	<input type="text" value="0"/>	4	There are distributed processing functions.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Internal Logical Files (ILFs)	<input type="text" value="0"/>	<input type="text" value="3"/>	<input type="text" value="0"/>	30	Performance is critical.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
External Interface Files (EIFs)	<input type="text" value="0"/>	<input type="text" value="25"/>	<input type="text" value="0"/>	175	The system runs in an existing, heavily utilized operational environment.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
<input type="button" value="Clear"/>					The system requires on-line data entry.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
					The on-line data entry requires transactions over multiple screens/operations.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
					ILFs are updated on-line.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
					The inputs, outputs, files or inquiries are complex.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
					The internal processing is complex.	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
					The code is designed to be reusable.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
					Conversions /installation are included in the design.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
					The system is designed for multiple installations in different organizations.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
					The system is designed to facilitate change and ease of use.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

<https://w3.cs.jmu.edu/bernstdh/web/common/webapps/oop/fpcalculator/FunctionPointCalculator.html>

- How much effort do you estimate, based upon this estimate (do not convert to LOC yet)?

$$\text{Effort} = \text{AFP} \times \text{PDR}$$

Where, PDR = Product delivery rate assuming 67.42% by taking 1000 LOC and 6 months of work

$$= 445 \times 0.6742$$

$$= 300$$

Reference:

<https://www.researchgate.net/publication/353366035> An Approach to Adjust Effort Estimation of Function Point Analysis Page 526

<https://melsatar.blog/2018/01/15/5-steps-to-software-development-effort-estimation/>

3. Assume Java is the language. How many LOC do you estimate?

LOC = Adjusted Function Point X Java SLOC/FP

= 445 X 53

= 23,585

Reference: Conversion of Function Points into LOC: Gearing Factors from slides

Effort<sub>Capers Jones</sub> = AFP/150 × AFP<sup>0.4</sup>

Effort = 445 / 150 × 445<sup>0.4</sup>

Effort = 2.96 X 11.46

Effort = 33.92

Reference:

[https://www.researchgate.net/publication/353366035\\_An\\_Approach\\_to\\_Adjust\\_Effort\\_Estimation\\_of\\_Function\\_Point\\_Analysis](https://www.researchgate.net/publication/353366035_An_Approach_to_Adjust_Effort_Estimation_of_Function_Point_Analysis) Page 527

1. Your boss listens nicely to your estimates and tells you they are 25% too large. How do you cut the effort by 25%? Hint: Change your design/plan so that there really is 25% less effort rather than working overtime or just saying a component is simpler. Think about the factors that are used specifically in the function point algorithm and the LOC algorithm. How do you reduce the expected LOC without cutting functionality?

Here are some ways to cut the effort by 25% without compromising the functionality:

Simplify the Design: Simplify the design of the system so that it requires less effort to develop. This can be done by removing unnecessary features, reducing the complexity of the system, and simplifying the user interface.

Reuse Existing Code: Reuse existing code and libraries rather than building everything from scratch. This can save a lot of development time and effort.

Optimize Code: Optimize the code to make it more efficient, reducing the number of lines of code required to implement the same functionality.

Automate Testing: Automate testing to reduce the amount of manual testing required, thereby saving time and effort.

Reference: <https://www.canva.com/graphs/block-diagrams/>  
<https://ifpug.org/>  
<https://www.javatpoint.com/software-engineering-functional-point-fp-analysis>  
<https://professionalqa.com/functional-point-analysis>  
<http://www.functionpointmodeler.com/fpm-infocenter/index.jsp?topic=%2Fcom.functionpointmodeler.fpm.help%2Feditafiles%2Fconcepts%2Fcon-11.html>  
<http://www.fredosaurus.com/notes-softeng/technology/fpa/fpa-effort.html>