

System Analysis and Design

ໂດຍ: ອຈ ສິມມິດ ທຸມມາລີ

ບົດທີ 5

ແບບຈຳລອງຂະບວນການ (Process Modeling)

ຈຸດປະສົງ

- ບອກຈຸດປະສົງຂອງແບບຈຳລອງໄດ້
- ສາມາດສ້າງແບບຈຳລອງເພື່ອນຳມາໃຊ້ກັບການວິເຄາະ ແລະ ອອກແບບລະບົບໄດ້
- ບອກຄວາມແຕກຕ່າງລະຫວ່າງແບບຈຳລອງແບບ Logical ແລະ ແບບ Physical ໄດ້
- ອະທິບາຍແນວຄິດການແຕກລະດັບຂອງແຜນວາດການໄຫຼຂໍ້ມູນໄດ້
- ສາມາດຊຽນຄຳອະທິບາຍການປະມວນຜົນໄດ້.

ຈຸດປະສົງຂອງແບບຈຳລອງ

- ເພື່ອຮຽນຮູ້ຂະບວນການສ້າງແບບຈຳລອງ
- ເພື່ອຫຼຸດຜ່ອນຄວາມຊ້ຳຊ້ອນໃນສິ່ງທີ່ເປັນນາມມະທຳໃຫ້ສາມາດເບິ່ງເຫັນໄດ້
- ຊ່ວຍຈົດຈຳລາຍລະອຽດທັງໝົດ
- ເພື່ອສື່ສານກັບສະມາຊິກໃນທີມພັດທະນາ
- ເພື່ອສື່ສານກັບຜູ້ໃຊ້ຕ່າງໆ ແລະ ກຸ່ມ Stakeholder
- ຊ່ວຍບັນທຶກຂໍ້ມູນໃນຮູບແບບເອກະສານເພື່ອປະໂຫຍດໃນການບຳລຸງຮັກສາລະບົບໃນອານາຄົດ.

ຊະນິດຂອງແບບຈຳລອງ

ໃນການພັດທະນາລະບົບຂ່າວສານ, ນັກວິເຄາະສາມາດນຳເອົາແບບຈຳລອງຊະນິດຕ່າງໆມາປະຍຸກໃຊ້ກັບງານພັດທະນາລະບົບ, ໂດຍແບບຈຳລອງແຕ່ລະຊະນິດກໍມີຈຸດເດັ່ນ ແລະ ການນຳສະເໜີມູມມອງຂອງລະບົບທີ່ແຕກຕ່າງກັນ ເຊັ່ນວ່າ ວຽກບາງຊະນິດສາມາດໃຊ້ແບບຈຳລອງທີ່ເປັນພຽງແຕ່ຄຳອະທິບາຍກໍສາມາດນຳໄປໃຊ້ງານໄດ້ແລ້ວ, ແຕ່ວຽກບາງຢ່າງ ພັດບໍ່ພຽງພໍ ດັ່ງນັ້ນ ຈຶ່ງຕ້ອງສ້າງແບບຈຳລອງທີ່ເປັນແຜນວາດເພື່ອໃຫ້ເຫັນພາບລວມຂອງລະບົບໄດ້ທັງໝົດ.

ແບບຈຳລອງທາງຄະນິດສາດ (Mathematical Models)

ເປັນຊຸດຂອງສູດຄິດໄລ່ທີ່ໃຊ້ອະທິບາຍລັກສະນະທາງເຕັກນິກຂອງລະບົບທີ່ສາມາດນຳມາໃຊ້ຢືນຢັນຄວາມຖືກຕ້ອງ ແລະ ທ່ຽງຕົງຂອງລະບົບໄດ້ເປັນຢ່າງດີອີກຍັງເປັນວິທີສຳລັບວຽກທາງດ້ານວິທະຍາສາດ ແລະ ວິສະວະກຳ ກັບການສະແດງຄວາມຕ້ອງການເລົ່ານີ້ອອກມາຜ່ານສູດການຄິດໄລ່, ນອກຈາກນີ້ຍັງສາມາດນຳມາໃຊ້ກັບຂະບວນການທຸລະກິດໄດ້ດີເຊັ່ນກັນ. ຕົວຢ່າງ ລະບົບເງິນເດືອນທີ່ຈຳເປັນຈະຕ້ອງສ້າງສູດຄິດໄລ່ຂຶ້ນມາເພື່ອຄິດໄລ່ເງິນເດືອນ ຫຼື ຄິດໄລ່ເງິນພາສີລາຍໄດ້ບຸກຄົນເປັນຕົ້ນ.

ແບບຈຳລອງທີ່ເປັນຄຳອະທິບາຍ (Descriptive Models)

ເປັນປະໂຫຍກ ຫຼື ຄຳເວົ້າທີ່ບັນລະຍາຍດ້ວຍພາສາທຳມະ
ຊາດ ເຊັ່ນວ່າ ການບັນທຶກຂໍ້ມູນການສຳພາດຂອງຜູ້ໃຊ້ເຖິງຄວາມ
ຕ້ອງການໃນດ້ານຕ່າງໆ ເຊິ່ງໃນບາງຄັ້ງຄຳອະທິບາຍເລົ່ານີ້ຖືເປັນວິທີ
ການບັນທຶກຂໍ້ມູນທີ່ດີທີ່ສຸດ ເຊັ່ນ ຄວາມຕ້ອງການຂອງຜູ້ໃຊ້ທີ່ຂຽນ
ຂຶ້ນບໍ່ເທົ່າໃດປະໂຫຍກກໍສາມາດນຳມາວິເຄາະເພື່ອສ້າງເປັນຂໍ້ກຳນົດ
ຄວາມຕ້ອງການຂຶ້ນມາລວມທັງຄຳອະທິບາຍເຖິງຂັ້ນຕອນການເຮັດ
ວຽກ ແລະ Pseudo code ທີ່ນຳມາໃຊ້ເພື່ອອອກແບບໂປຣ
ແກຣມເປັນຕົ້ນ.

ແບບຈຳລອງແຜນວາດ (Graphical Models)

ເປັນແບບຈຳລອງທີ່ມີປະໂຫຍດຫຼາຍສຳລັບງານພັດທະນາລະບົບ ເຊິ່ງຖືກສ້າງຂຶ້ນໂດຍນັກວິເຄາະລະບົບໂດຍແບບຈຳລອງປະເພດນີ້ປະກອບດ້ວຍແຜນສະແດງ ຫຼື Diagram ຕ່າງໆທີ່ນຳສະເໜີພາບລວມຂອງລະບົບທີ່ມີຄວາມຊັບຊ້ອນໃຫ້ສາມາດສື່ສານລະຫວ່າງກັນໄດ້ງ່າຍຂຶ້ນ ຫຼື ອາດເຂົ້າໃຈໄດ້ໂດຍທັນທີ. ເຊັ່ນ: ແຜນວາດການໄຫຼຂໍ້ມູນ (Data Flow Diagram ຫຼື DFD), ແຜນວາດຄວາມສຳພັນຂອງຂໍ້ມູນໃນຖານຂໍ້ມູນ (Entity Relationship ຫຼື ER) ເປັນຕົ້ນ.

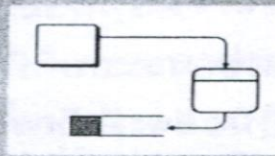
ແບບຈຳລອງທີ່ນຳມາໃຊ້ໃນການວິເຄາະລະບົບ

ການກຳນົດຄວາມຕ້ອງການເປັນກົດຈະກຳໜຶ່ງໃນໄລຍະ
ການວິເຄາະທີ່ກ່ຽວຂ້ອງກັບການສ້າງແບບຈຳລອງ Logical
(Logical Model) ເພື່ອນຳສະເໜີລາຍລະອຽດໃນສິ່ງທີ່ຕ້ອງການ
ວ່າມີຫຍັງແດ່, ໂດຍແບບຈຳລອງດັ່ງກ່າວນີ້ບໍ່ໄດ້ຂຶ້ນກັບເຕັກໂນໂລ
ຊີໃດໆ

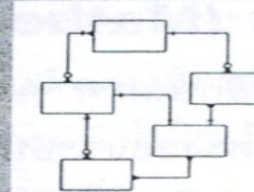
ຕົວຢ່າງແບບຈຳລອງທີ່ຖືກສ້າງຂຶ້ນໃນລະຫວ່າງການວິເຄາະ

1. buy new car
2. sell car
3. get car serviced
4. make payment
5. trade in car

Event List



Data Flow Diagram (DFD)



Entity-Relationship Diagram (ERD)

dataflow 1 =
element 1+
element 2+
element 3

Data Flow Definition

element 1 =
description
data type
validation rules

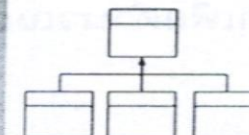
Data Element Definition

do this
if ...
else ...
while x
do that
do the other

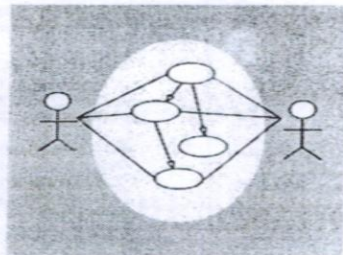
Process Description



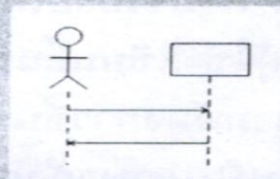
Location Diagram



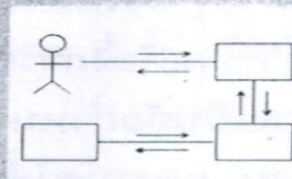
Class Diagram



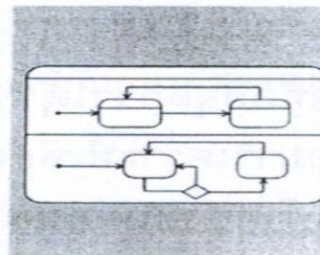
Use Case Diagram



Sequence Diagram



Collaboration Diagram

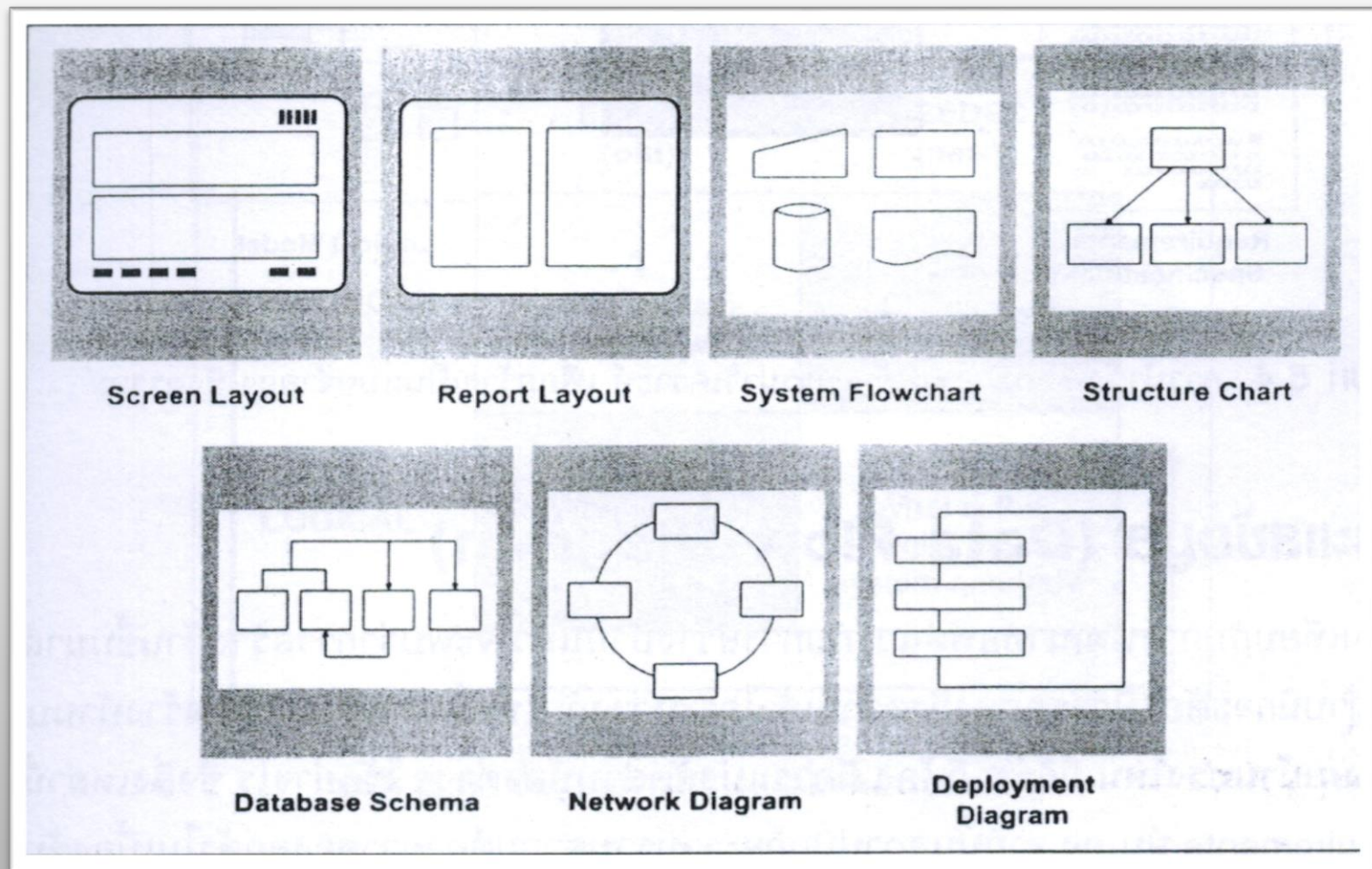


Statechart Diagram

ແບບຈຳລອງທີ່ນຳມາໃຊ້ໃນການອອກແບບລະບົບ

ແບບຈຳລອງ Physical (Physical Model) ຈະຖືກສ້າງຂຶ້ນໃນໄລຍະອອກແບບ ໂດຍແບບຈຳລອງດັ່ງກ່າວເຮັດໃຫ້ເຫັນຮູບຮ່າງໜ້າຕາບາງສ່ວນຂອງລະບົບ ວ່າຖືກສ້າງຂຶ້ນມາໄດ້ແບບໃດພາຍໃຕ້ເຕັກໂນໂລຊີນັ້ນໆ. ແຕ່ເຖິງຢ່າງໃດກໍຕາມ ແບບຈຳລອງ Physical ບາງສ່ວນເປັນສ່ວນຂະຫຍາຍທີ່ໄດ້ມາຈາກແບບຈຳລອງທີ່ຖືກສ້າງຂຶ້ນໃນລະຫວ່າງການວິເຄາະ, ໃນຂະນະດຽວກັນ ບາງແບບຈຳລອງອາດຈະຖືກນຳມາໃຊ້ທັງການວິເຄາະ ແລະ ການອອກແບບ ເຊັ່ນວ່າ Class Diagram ເປັນຕົ້ນ.

ຕົວຢ່າງແບບຈຳລອງທີ່ຖືກສ້າງຂຶ້ນໃນລະຫວ່າງການອອກແບບ



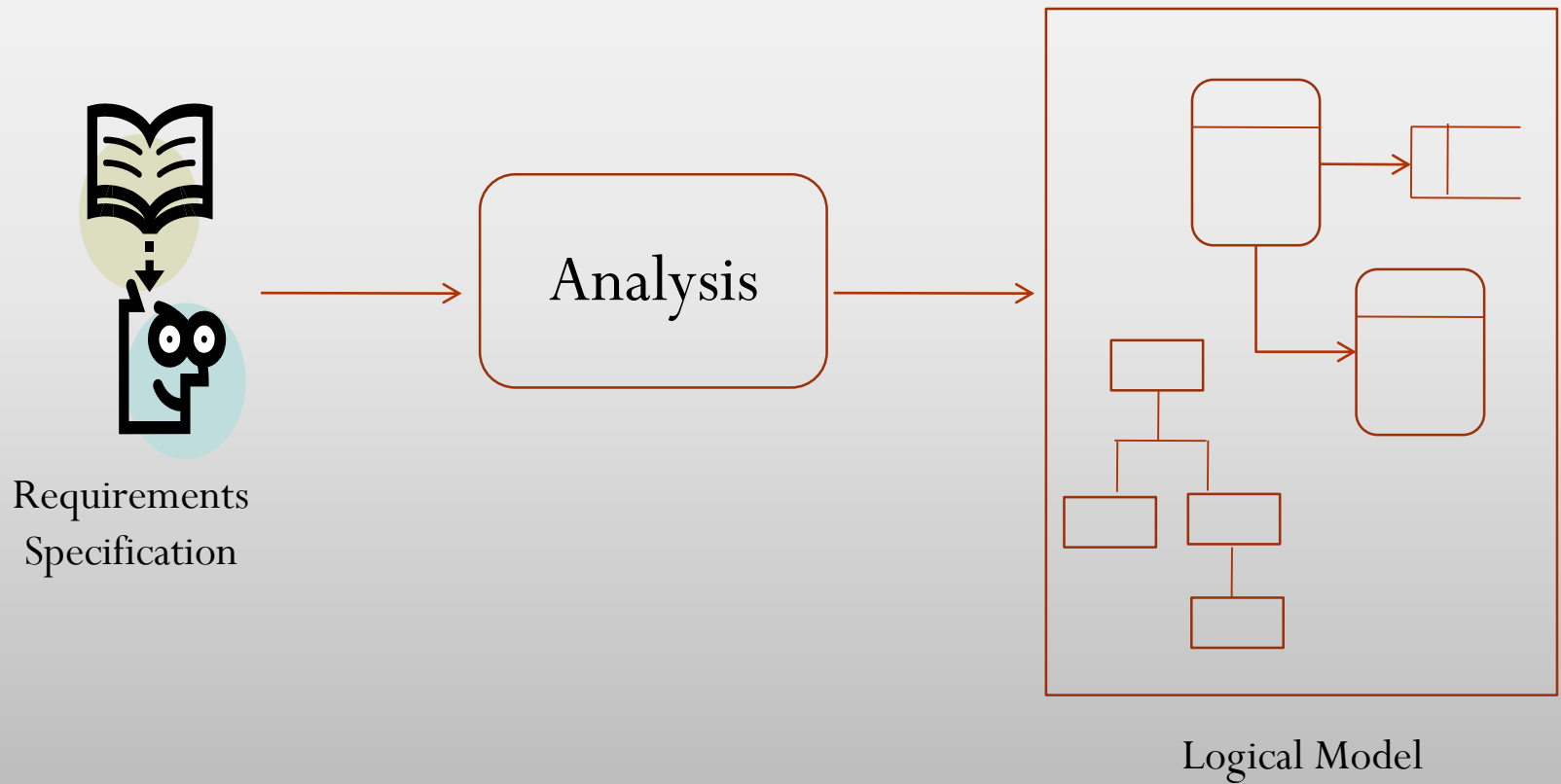
ແບບຈຳລອງຂະບວນການ (Process Model)

ແບບຈຳລອງຂະບວນການ ເປັນແຜນວາດທີ່ນຳມາໃຊ້ແທນຟັງຊັນການທຳງານ ທີ່ກ່ຽວຂ້ອງກັບຂະບວນການ ດ້ວຍການຈັບໃຈຄວາມສຳຄັນໃນເລື່ອງການຈັດການ, ການຈັດເກັບ ແລະ ການກະຈາຍຂໍ້ມູນ ລະຫວ່າງແບບຈຳລອງກັບສະພາບແວດລ້ອມລວມເຖິງອົງປະກອບພາຍໃນລະບົບ ເຊິ່ງແບບຈຳລອງດັ່ງກ່າວ ເອີ້ນວ່າແຜນວາດການໄຫຼຂໍ້ມູນ.

ແຜນວາດການໄຫຼຂໍ້ມູນ.

ແຜນວາດການໄຫຼຂໍ້ມູນ ເປັນແບບຈຳລອງທີ່ຖືກນຳມາໃຊ້ກັບ ວິທີການພັດທະນາລະບົບ ຕາມແນວທາງການວິເຄາະ ແລະ ອອກບລະບົບແບບໂຄງສ້າງ ແລະ ຖືເປັນໜຶ່ງໃນເຕັກນິກທີ່ ຖືກນຳມາໃຊ້ໃນການພັດທະນາລະບົບແບບໂຄງສ້າງຫຼາຍທີ່ ສຸດ, ສ່ວນ ການພັດທະນາແບບວັດຖຸຈະນຳໃຊ້ ເຕັກນິກ Use Case Diagram.

ແບບຈຳລອງຂະບວນການ (Process Model)




ແຜນວາດການໄຫຼຂໍ້ມູນ

ແຜນວາດການໄຫຼຂໍ້ມູນ (Data Flow Diagram: DFD) ເປັນ ແບບຈຳລອງຂະບວນການທີ່ນຳມາໃຊ້ກັບການວິເຄາະ ແລະ ອອກແບບ ລະບົບແບບໂຄງສ້າງ, ໂດຍແຜນວາດດັ່ງກ່າວນີ້ໃຊ້ເປັນເຄື່ອງມືໃນການ ພັດທະນາລະບົບ ແລະ ສະແດງຄວາມສຳພັນລະຫວ່າງຂະບວນການ ແລະ ຂໍ້ມູນທີ່ກ່ຽວຂ້ອງ, ເຊິ່ງຂໍ້ມູນຢູ່ໃນແຜນວາດຈະເຮັດໃຫ້ຮູ້ເຖິງ ຂໍ້ມູນມາຈາກ ໃສ, ຂໍ້ມູນໄປທາງໃດ ແລະ ເກີດເຫດການໃດກັບຂໍ້ມູນໃນລະຫວ່າງການ ໄຫຼ.

ຂັ້ນຕອນການສ້າງແຜນວາດການໄຫຼຂໍ້ມູນ

	Exists (old)	Needed (new)
Physical	1 What system exists now	3 Create the system needed
Logical	2 What is the “Logic” of what exists ?	4 What is the “Logic” of the system needed ?



ຕົວຢ່າງ Logical ແລະ Physical

ສົມມຸດວ່າເຮົາໄປຊື້ສິນຄ້າຢູ່ຫ້າງສັບພະສິນຄ້າ, ເມື່ອເລືອກສິນຄ້າແລ້ວ ກໍໄປ ສຳລະເງິນ. ຄຳວ່າ ສຳລະເງິນ ແມ່ນ Logical, ໃນຂະນະດຽວກັນການສຳລະເງິນກໍມີຫຼາຍຮູບແບບເຊັ່ນ ສຳລະເງິນສົດ ຫຼື ສຳລະເງິນຜ່ານບັດເຄຣດິດ ດັ່ງນັ້ນ ລາຍລະອຽດຂອງການສຳລະເງິນນີ້ ແມ່ນ Physical. ສະນັ້ນຈຶ່ງສະຫຼຸບໄດ້ວ່າ Logical ບໍ່ໄດ້ເນັ້ນລາຍລະອຽດພຽງໃຫ້ຮູ້ວ່າຈະເຮັດຫຍັງ(What), ສ່ວນ Physical ຈະເນັ້ນລາຍລະອຽດວ່າ ເຮັດແນວໃດ(How).

ສະຫຼຸບ


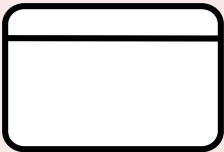



ເມື່ອແຜນວາດການໄຫຼຂໍ້ມູນເປັນແບບຈຳລອງ Logical ຈຶ່ງສະແດງຂະບວນການ ແລະ ຂໍ້ມູນທີ່ກ່ຽວຂ້ອງພາຍໃນລະບົບ ດ້ວຍການມັງເນັ້ນວ່າລະບົບຕ້ອງເຮັດຫຍັງເປັນສຳຄັນ ໂດຍມູມມອງຂອງຜູ້ໃຊ້ຕາມກຸ່ມຕ່າງໆຈະໃຊ້ປະໂຫຍດຈາກແຜນວາດນີ້ເພື່ອ:

- ມູມມອງຂອງລູກຄ້າ ຫຼື ຜູ້ໃຊ້ງານ: ໃຊ້ປະໂຫຍດເພື່ອສະແດງພາບລວມຂອງລະບົບ
- ມູມມອງຂອງນັກຂຽນໂປຣແກຣມ: ໃຊ້ປະໂຫຍດເພື່ອສະແດງລາຍລະອຽດຂອງລະບົບ ແລະ ໃຊ້ເປັນແນວທາງໃນການຂຽນໂປຣແກຣມ
- ມູມມອງຂອງນັກວິເຄາະລະບົບ: ໃຊ້ເພື່ອສະແດງພາບລວມຂອງລະບົບ ແລະ ລາຍລະອຽດ.

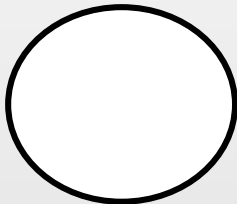
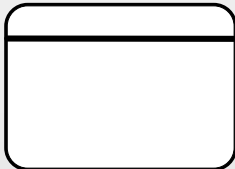




ຈຸດປະສົງຂອງແຜນວາດການໄຫຼຂໍ້ມູນ

1. ເປັນແຜນວາດທີ່ສະຫຼຸບລວມຂໍ້ມູນທັງໝົດທີ່ໄດ້ຈາກການວິເຄາະໃນຮູບແບບຂອງການພັດທະນາແບບໂຄງສ້າງ
2. ເປັນຂໍ້ຕົກລົງຮ່ວມກັນລະຫວ່າງນັກວິເຄາະລະບົບກັບຜູ້ໃຊ້ງານ
3. ເປັນແຜນສະແດງທີ່ນຳໄປໃຊ້ປະໂຫຍດຕໍ່ໃນຂັ້ນຕອນການອອກແບບ
4. ເປັນແຜນສະແດງທີ່ໃຊ້ໃນການອ້າງອີງ ຫຼື ເພື່ອໃຊ້ສຳລັບການປັບປຸງ ແລະ ພັດທະນາຕໍ່ໃນອານາຄົດ
5. ຮູ້ທີ່ມາ ແລະ ທີ່ໄປຂອງຂໍ້ມູນທີ່ໄຫຼໄປຍັງຂະບວນການຕ່າງໆ

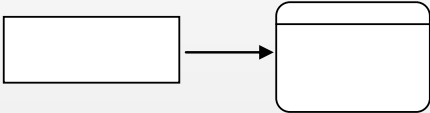
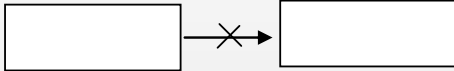
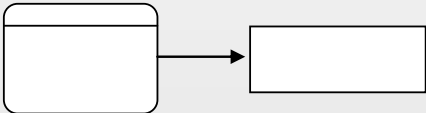
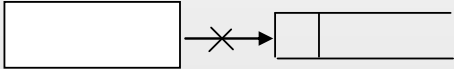
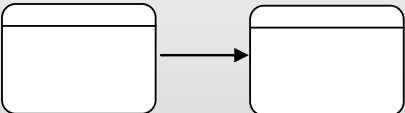
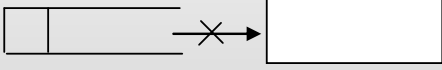
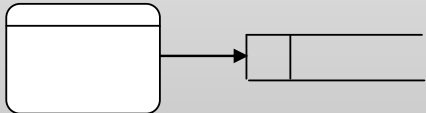
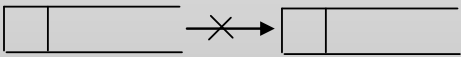
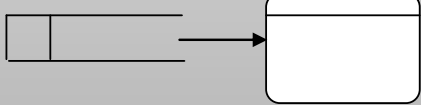
ສັນຍາລັກທີ່ໃຊ້ໃນແຜນວາດການໄຫຼຂໍ້ມູນ

ລຸ້ນ	ສັນຍາລັກ	ຄວາມໝາຍ
Boundary Or External Entity		ຊອບເຂດ ໝາຍເຖິງພາສ່ວນທີ່ກ່ຽວຂ້ອງກັບລະບົບເຊິ່ງລະບົບບໍ່ສາມາດຄວບຄຸມໄດ້
Process		ປະມວນຜົນ ຫຼື ໜ້າວຽກທີ່ຈະເຮັດໃນໂຄງການນັ້ນໆ
Data Store		ບ່ອນຈັດເກັບຂໍ້ມູນ
Data Flow		ການໄຫຼຂອງຂໍ້ມູນ
Real-Time Link		ການເຊື່ອມໂຍງໄລຍະໄກທີ່ມີການຕອບກັບແບບທັນທີທັນໃດ

ສັນຍາລັກທີ່ໃຊ້ໃນແຜນວາດການໄຫຼຂໍ້ມູນ

Yourdon/Demarco	Gane and Sarson
	
	
	

ຫຼັກການການຂຽນແຜນວາດການໄຫຼຂໍ້ມູນ

ອານຸຍາດ	ບໍ່ອານຸຍາດ
	
	
	
	
	

ປຽບທຽບແຜນວາດການໄຫຼຂໍ້ມູນທີ່ບໍ່ຖືກຕ້ອງ ແລະ ຖືກຕ້ອງ

Process:

ເມື່ອມີຂໍ້ມູນເຂົ້າໄປທີ່ Process ກໍຕ້ອງມີຂໍ້ມູນ ຫຼື ຜົນຮັບອອກຈາກ Process ເຊັ່ນກັນ ຄືຈະເປັນໄປບໍ່ໄດ້ທີ່ມີສະເພາະຂໍ້ມູນເຂົ້າຢ່າງດຽວ ຫຼື ອອກຢ່າງດຽວ ດັ່ງຮູບ (a), (b) ແລະ process ຈະຕ້ອງໃຊ້ຄໍາກໍານະເທົ່ານັ້ນ

Data store:

- ຂໍ້ມູນຈະໄຫຼຈາກ Data store ໜຶ່ງໄປຫາ Data store ໜຶ່ງໂດຍກົງບໍ່ໄດ້ ຈະຕ້ອງຜ່ານ Process ເທົ່ານັ້ນ ດັ່ງຮູບ (c)
- ຂໍ້ມູນທີ່ສົ່ງຜ່ານຈາກ External Entity ບໍ່ສາມາດໄຫຼເຂົ້າໄປ Data store ໂດຍກົງໄດ້ຈະຕ້ອງໃຊ້ Process ເປັນຕົວກາງໃນການເຊື່ອມໂຍງເພື່ອຈັດເກັບຂໍ້ມູນໃນ Data store ຕໍ່ໄປ ດັ່ງຮູບ (d)
- ຂໍ້ມູນທີ່ໄຫຼຜ່ານຈາກ Data store ບໍ່ສາມາດເຊື່ອມໂຍງເຂົ້າກັບ External Entity ໄດ້ໂດຍກົງຈະຕ້ອງຜ່ານ Process ເທົ່ານັ້ນ ດັ່ງຮູບ (e)
- Data store ຈະຕ້ອງໃຊ້ຄໍານາມ

ປຽບທຽບແຜນວາດການໄຫຼຂໍ້ມູນທີ່ບໍ່ຖືກຕ້ອງ ແລະ ຖືກຕ້ອງ

External Entity:

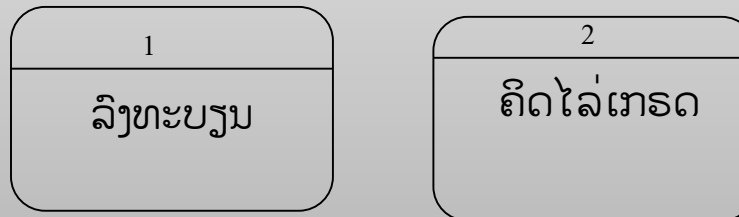
External Entity ບໍ່ສາມາດເຊື່ອມໂຍງຂໍ້ມູນຫາກັນໄດ້ ຈະຕ້ອງໃຊ້ Process ເປັນຕົວກາງເພື່ອການສົ່ງຜ່ານ ດັ່ງຮູບ (f) ແລະ ຊື່ ຂອງ External Entity ຈະໃຊ້ຄຳນາມເທົ່ານັ້ນ

Data flow:

- ການໄຫຼຂໍ້ມູນທີ່ມີຫົວລູກສອນຊື່ໄປທີ່ Process ໝາຍເຖິງ Process ມີການອ່ານ ຫຼື ການດຶງຂໍ້ມູນຈາກ Data store ມາໃຊ້ວຽກ
- ການໄຫຼຂໍ້ມູນຈາກ Process ທີ່ມີຫົວລູກສອນຊື່ໄປຍັງ Data store ໝາຍເຖິງການ Update ຫຼື ການເພີ່ມຂໍ້ມູນລົງໄປທີ່ Data store
- ການໄຫຼຂໍ້ມູນທີ່ມີຫົວລູກສອນທັງສອງດ້ານທີ່ເຊື່ອມໂຍງລະຫວ່າງ Process ກັບ Data store ໝາຍເຖິງມີການດຶງຂໍ້ມູນຈາກ Data store ມາປັບປຸງ ແລະ ມີການ Update ຂໍ້ມູນລົງໃນ Data store
- ການໄຫຼຂໍ້ມູນບໍ່ສາມາດຢ້ອນກັບໄປຍັງ Process ເດີມໄດ້ ຢ່າງໜ້ອຍຕ້ອງເຊື່ອມໂຍງຜ່ານ Process ໜຶ່ງເພື່ອສົ່ງຜ່ານຍ້ອນກັບມາຍັງ Process ເດີມ ດັ່ງຮູບ (g) ແລະ ຊື່ທີ່ລະບຸໃນການໄຫຼຂໍ້ມູນຈະໃຊ້ຄຳນາມ

Process

ແມ່ນຂະບວນການທີ່ຕ້ອງເຮັດໃນລະບົບ ແລະ ການຂຽນ Process ຕ້ອງຂຽນ ເປັນຄໍາກຳມະເຊັ່ນ: ລົງທະບຽນ, ຖອນວິຊາຮຽນ, ເພີ່ມວິຊາຮຽນ ແລະ ລາຍງານ ເປັນຕົ້ນ. ຈຳນວນ Process ຄວນຈະຢູ່ລະຫວ່າງ 2 ເຖິງ 7 Process ບໍ່ຄວນໜ້ອຍ ຫຼື ຫຼາຍເກີນໄປ. ຖ້າຫຼາຍເກີນໄປ ຈະເຮັດໃຫ້ແຜນວາດການໄຫຼຂໍ້ມູນເຂົ້າໃຈຍາກເນື່ອງຈາກມີຄວາມຊັບຊ້ອນຫຼາຍແຕ່ຖ້າໜ້ອຍເກີນໄປຈະ ເຮັດໃຫ້ລາຍລະອຽດຂອງແຜນວາດການໄຫຼຂໍ້ມູນບໍ່ພຽງພໍ. ໝາຍ ເລກ Process ທີ່ຂຽນຢູ່ດ້ານເທິງ ຂອງສັນຍາລັກ Process ບໍ່ໝາຍເຖິງການເຮັດວຽກຕາມລຳດັບຂອງ Process ແຕ່ໝາຍເຖິງໃຫ້ຮູ້ວ່າ ແມ່ນ Process ໃດເທົ່ານັ້ນ ແລະ Process ບໍ່ສາມາດຊ້ຳກັນໄດ້.

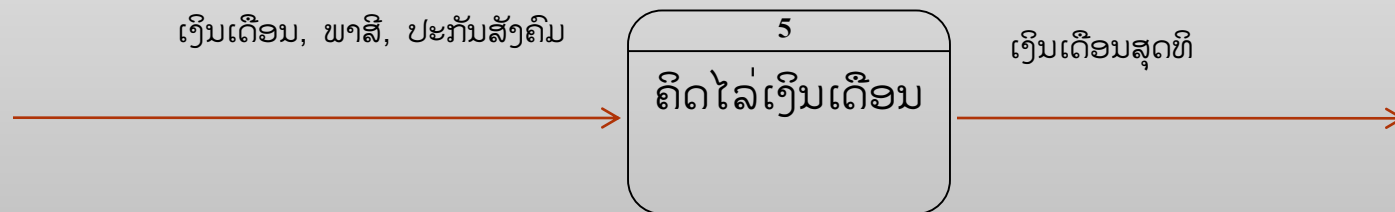


Process

Process ໃນແຜນວາດການໄຫຼຂໍ້ມູນຈະບໍ່ສະແດງລາຍລະອຽດກ່ຽວກັບການເຮັດວຽກ ດັ່ງນັ້ນ Process ຈຶ່ງເປັນເໝືອນກ່ອງດຳ(Black Bok) ທີ່ນຳສະໜີພຽງວ່າເຮັດຫຍັງ ໂດຍມີ Data Flow ຫຍັງແດ່ທີ່ input ເຂົ້າມາ ແລະ ມີ Data Flow ຫຍັງແດ່ທີ່ Output ອອກໄປ, ສ່ວນລາຍລະອຽດກ່ຽວກັບການເຮັດວຽກຂອງ Process ຈະປະກົດຢູ່ໃນແບບຈຳລອງຄຳອະທິບາຍ ຫຼື Process Description

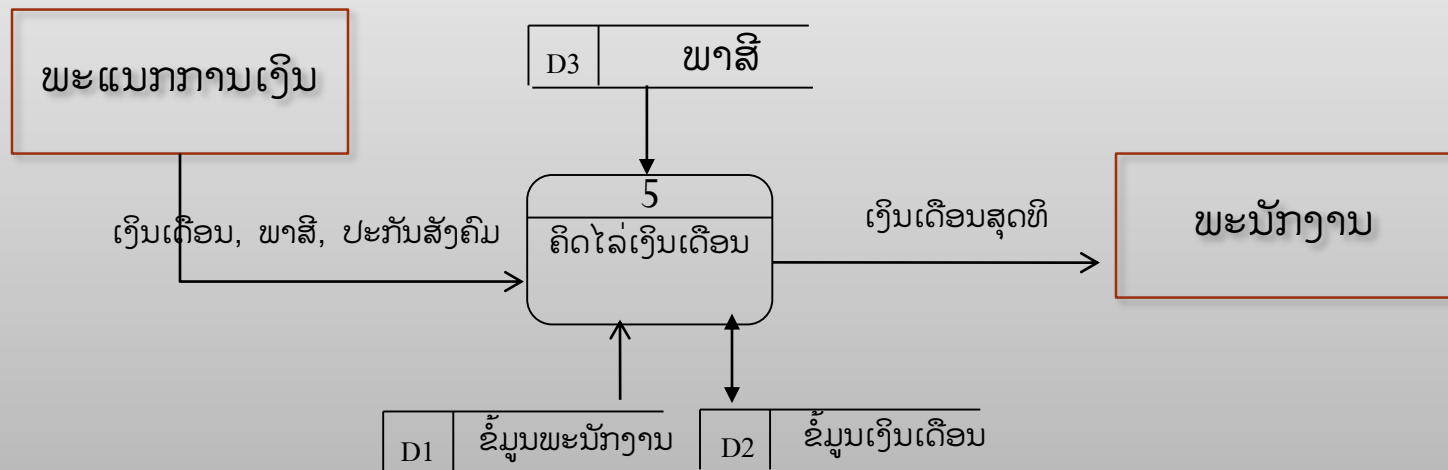
ຕົວຢ່າງການເຮັດວຽກຂອງ Process

Data Flow ທີ່ອອກຈາກ Process ຈະເຮັດໃຫ້ເກີດມີການປ່ຽນແປງຂໍ້ມູນ. ຈາກຕົວຢ່າງ ພົບວ່າ Data Flow ເງິນເດືອນ, ພາສີ ແລະ ປະກັນສັງຄົມ ໄດ້ Input ໄປຍັງ Process ໝາຍເລກ 5 ເຊິ່ງແທນກິດຈະກຳ ຄິດໄລ່ເງິນເດືອນ ແລະ ເມື່ອ Process ດັ່ງກ່າວປະມວນຜົນແລ້ວ ກໍຈະມີຂໍ້ມູນ Output ອອກມາ ເປັນເງິນເດືອນສຸດທິ.




Data Flow

ແມ່ນເສັ້ນທາງທີ່ຂໍ້ມູນຈະເຄື່ອນໄປເຮັດໃຫ້ສາມາດຮູ້ເຖິງຂໍ້ມູນຕ່າງໆທີ່ເຄື່ອນໄຫວໄປມາລະຫວ່າງ Process, Datastore ແລະ Boundary. ສິ່ງທີ່ຈະຕ້ອງຈຳກັດທຸກໆ Process ໃນແຜນວາດການໄຫຼຂໍ້ມູນເມື່ອມີ Data flow ທີ່ Input ເຂົ້າໄປ ກໍຈະຕ້ອງມີ Data flow Output ອອກມາສະເໝີ ບໍ່ວ່າຈະເປັນການ Input ຫຼື Output ມາຈາກ Process, ຈາກ Data Store ຫຼື ຈາກ External Entity ກໍຕາມ.



External Entities

External Entities ເປັນພາກສ່ວນທີ່ຢູ່ນອກລະບົບເຮັດໜ້າທີ່ສົ່ງຂໍ້ມູນເຂົ້າມາ
ຍັງ Process ເພື່ອສະແດງເຖິງແຫຼ່ງທີ່ມາຂອງຂໍ້ມູນ (Source) ລວມເຖິງຮັບເອົາຂໍ້
ມູນຈາກ Process ເພື່ອສະແດງເຖິງຈຸດສິ້ນສຸດຂອງການສົ່ງຂໍ້ມູນ (Sink) ແລະ
ດ້ວຍເຫດນີ້ເອງຈຶ່ງເຮັດໃຫ້ສັນຍາລັກນີ້ສາມາດເອີ້ນໄດ້ແບບເຊັ່ນ: Source, Sink,
Destination, External Agent ຫຼື Boundary ລ້ວນແຕ່ມີຄວາມໝາຍດຽວກັນ



ນັກສຶກສາ

ນັກສຶກສາ

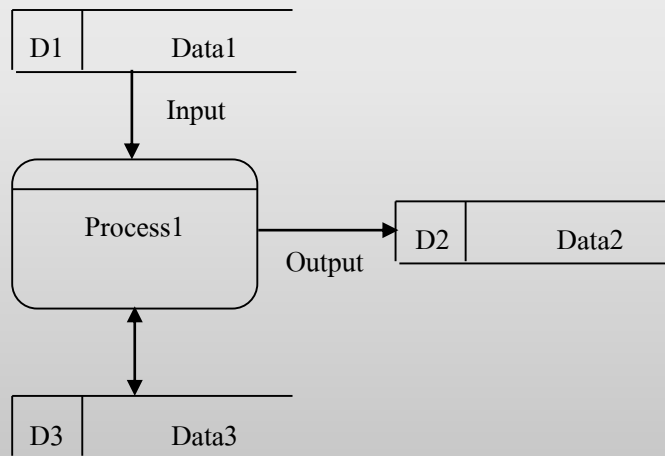
External Entities

External Entities ສາມາດເປັນໄດ້ທັງບຸກຄົນ, ໜ່ວຍງານ ຫຼື ລະບົບງານ ເຊິ່ງໃນການພິຈາລະນາວ່າອັນໃດເປັນ Boundary ທີ່ກ່ຽວຂ້ອງກັບລະບົບນັ້ນ ຈະພິຈາລະນາເຖິງບຸກຄົນທີ່ລະບົບບໍ່ສາມາດຄວບຄຸມໄດ້ ເຊັ່ນ : ລະບົບຮ້ານເຊົ່າ DVD ຈະມີ External Entities ຄື: ລູກຄ້າ, ຜູ້ຈັດການ ແລະ ສຳນັກຫັກບັດເຄຣດິດ. ແຕ່ບາງຄົນອາດຈະເອົາພະນັກງານມາເປັນ External Entities ນຳ, ໃນຄວາມເປັນຈິງແມ່ນບໍ່ຖືກຕ້ອງເນື່ອງຈາກບຸກຄົນໃດທີ່ປະຕິບັດກັບ Process ໂດຍກົງຈະຖືວ່າເປັນສ່ວນໜຶ່ງຂອງຂະບວນການ ແລະ ຜູ້ປະຕິບັດຕາມຂະບວນການຈະມີການອະທິບາຍລາຍລະອຽດລົງໃນຄຳອະທິບາຍການປະມວນຜົນຂອງຂໍ້ມູນ, ແຕ່ຈະບໍ່ປະກົດໃນແຜນວາດການໄຫຼຂໍ້ມູນ.

ເຖິງຢ່າງໃດກໍຕາມກໍມີຂໍ້ຍົກເວັ້ນສຳລັບບຸກຄົນພາຍໃນລະບົບທີ່ຖືວ່າເປັນ External Entities ໂດຍທຳມະຊາດ ເຊັ່ນ: ຜູ້ຈັດການ ຫຼື ທີມງານເປັນຕົ້ນ.

Data Store

Data Store ເປັນແຫຼ່ງເກັບຂໍ້ມູນເຊິ່ງຈະບໍ່ສົນໃຈວ່າລະບົບຈະໃຊ້ສີ່ຈັດເກັບຂໍ້ມູນປະເພດໃດກໍຕາມ. ທຸກໆ Data store ຈະມີຊື່ຂໍ້ມູນຈັດເກັບ ແລະ ມີເລກລຳດັບເຊັ່ນ D1, D2, D3,.....ເຊິ່ງ Data store ນີ້ຈະຖືກໃຊ້ງານໂດຍ Process



ໝາຍເຫດ: Data Store ສາມາດຊື່າກັນໄດ້

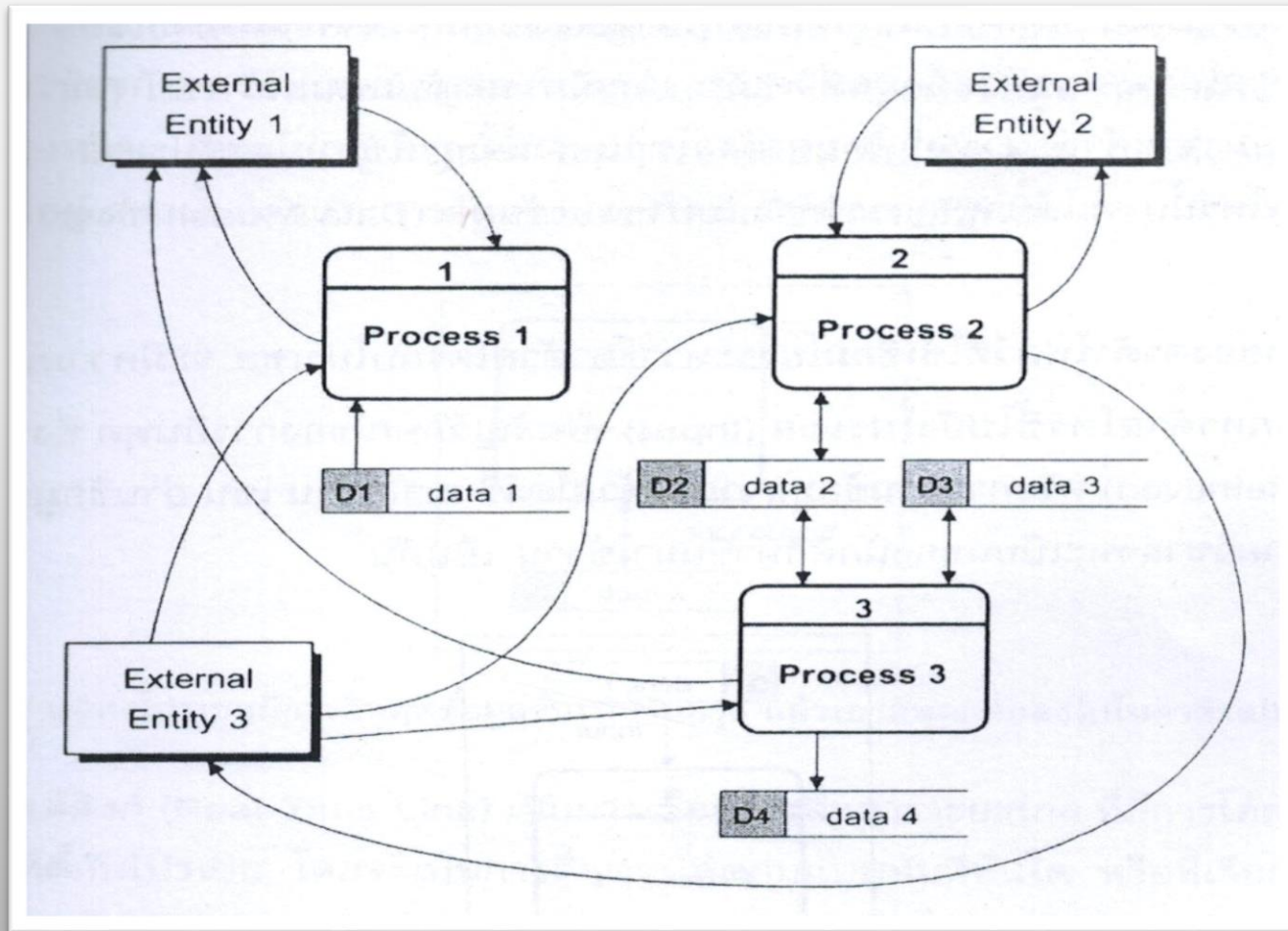
Real-Time Link

ເປັນການເຊື່ອມໂຍງສື່ສານໄລຍະໃກທີ່ມີການໂຕ້ຕອບກັນໄປມາລະຫວ່າງ External Entity ກັບ Process ໂດຍຈະເປັນການສື່ສານໂຕ້ຕອບທັນທີທັນໃດ ຫຼື ເອີ້ນກັນວ່າ Real Time Link.

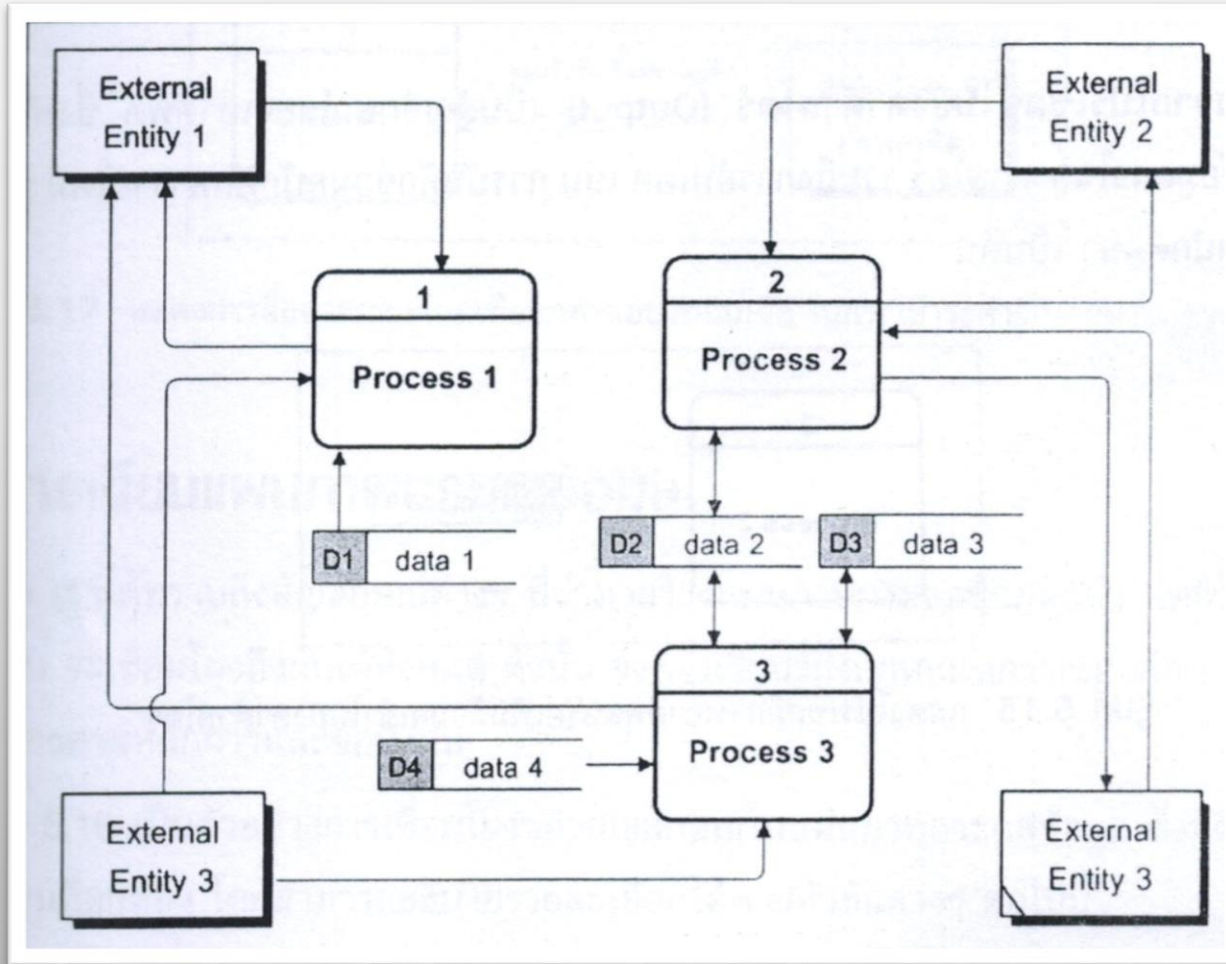
ຕົວຢ່າງ: ການກວດສອບບັດເຄດິດທະນາຄານເພື່ອລໍການອານຸມັດສິນເຊື່ອຈາກສຳນັກຫັກກັບນຊີບັດເຄດິດເປັນຕົ້ນ



ແຜນວາດທີ່ມີ Data Flow ເຊື່ອມໂຍງຂ້າມໄປມາບໍ່ເປັນລະບຽບ



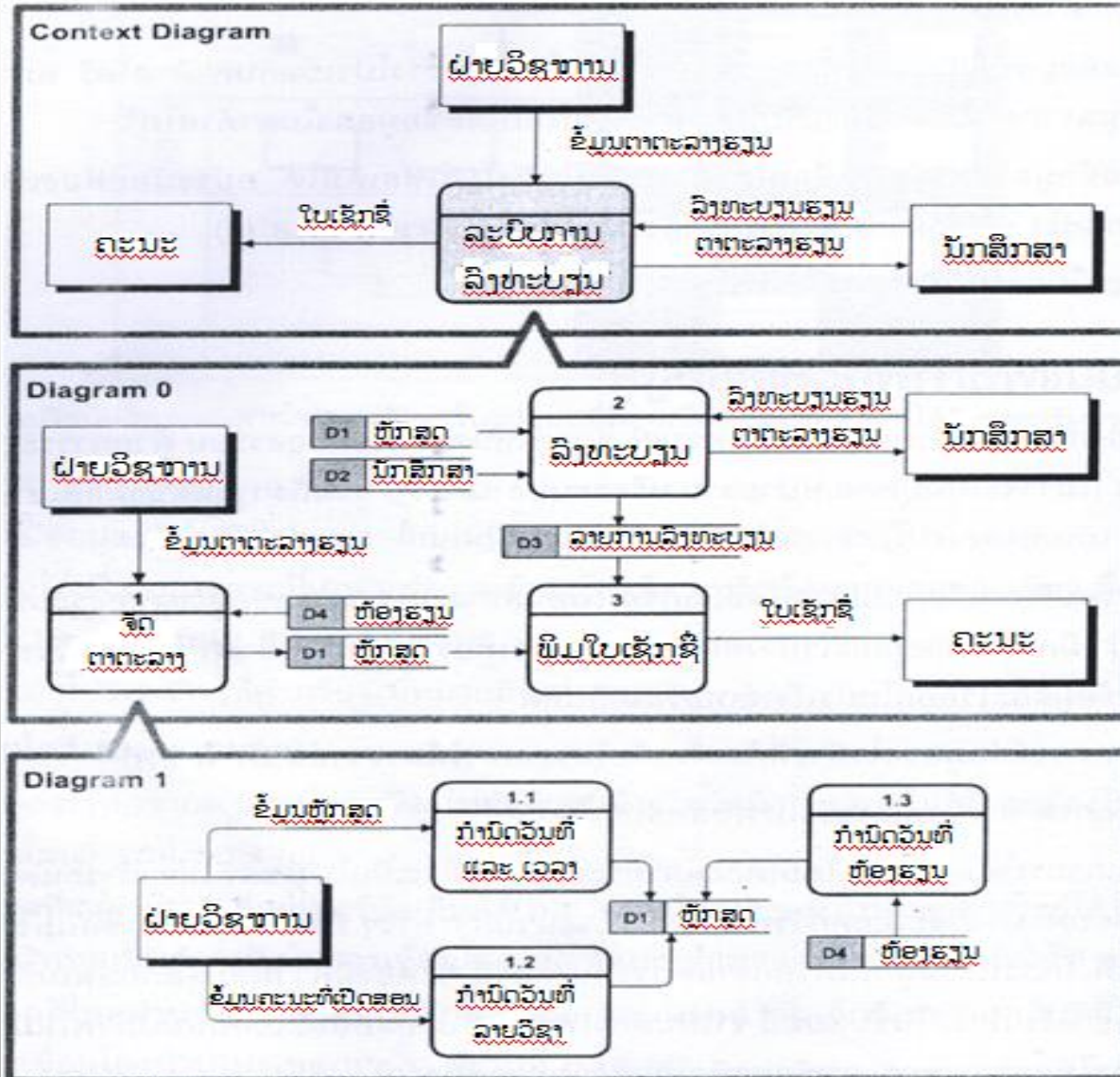
ແຜນວາດທີ່ມີ Data Flow ເຊື່ອມໂຍງເປັນລະບຽບ



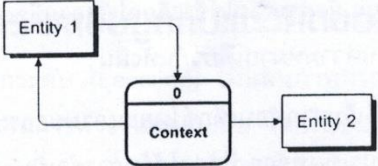
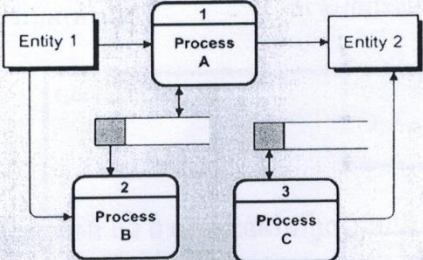
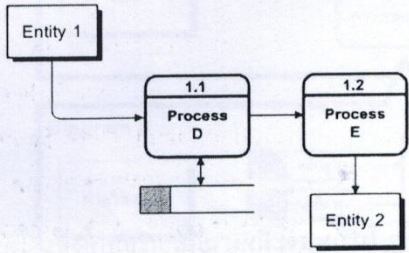
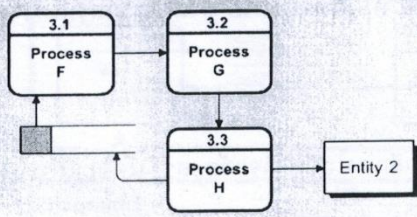
ຂັ້ນຕອນການຂຽນແຜນວາດການໄຫຼຂໍ້ມູນ

- ວິເຄາະໃຫ້ໄດ້ວ່າໃນລະບົບມີ External Entities ໃດທີ່ກ່ຽວຂ້ອງກັບລະບົບ
- ອອກແບບແຜນວາດເນື້ອຫາ (Context Diagram)
- ວິເຄາະຂໍ້ມູນໃນລະບົບວ່າຄວນມີຂໍ້ມູນໃດແດ່ທີ່ຄວນຈັດເກັບ
- ວິເຄາະຂະບວນການວ່າມີ Process ຫຼັກ ແລະ Process ຍ່ອຍໃດແດ່
- ຂຽນແຜນວາດການໄຫຼຂໍ້ມູນໃນລະດັບຕ່າງໆ
- ກວດສອບ, ແກ້ໄຂຈົນໄດ້ແຜນວາດທີ່ສົມບູນ ແລະ ຖືກຕ້ອງ.

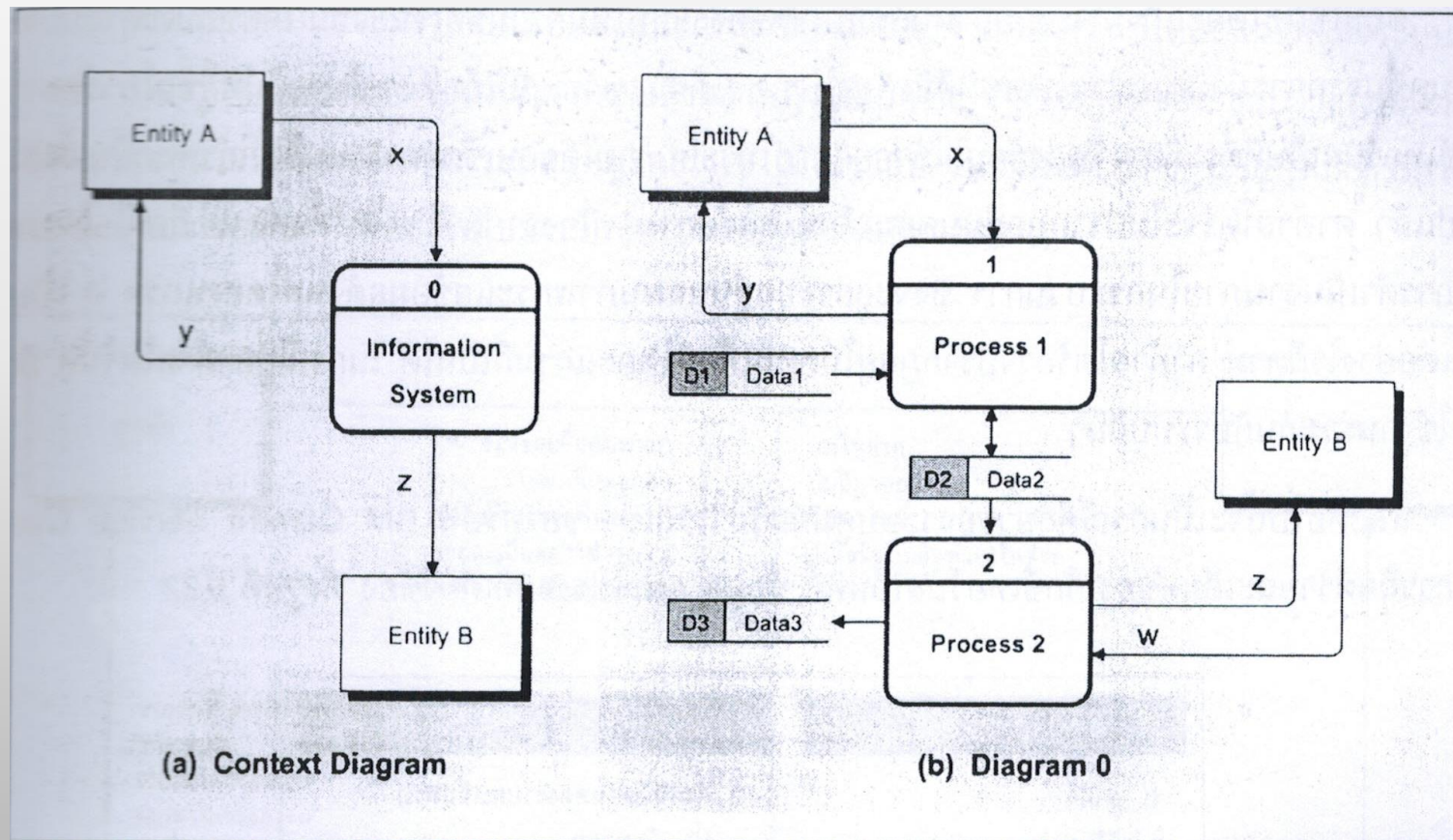
ແຜນວາດການໄຫຼຂໍ້ມູນກັບແນວຄິດການແຕກລະດັບ



ປຽບທຽບການແຕກລະດັບດ້ວຍການອ້າງອີງແບບ Level ກັບແບບ Numbering Technique

(Data Flow Diagram)	Level	Numbering Technique
	Context Diagram (Level 0)	Context Diagram
	DFD-Level 1 (Top Level)	Diagram 0
	DFD-Level 2 of Process A	Diagram 1
	DFD-Level 2 of Process C	Diagram 3

ການກວດສອບຄວາມສົມດູນຂອງແຜນວາດການໄຫຼຂໍ້ມູນ



ແຜນວາດເນື້ອຫາ (Context Diagram)

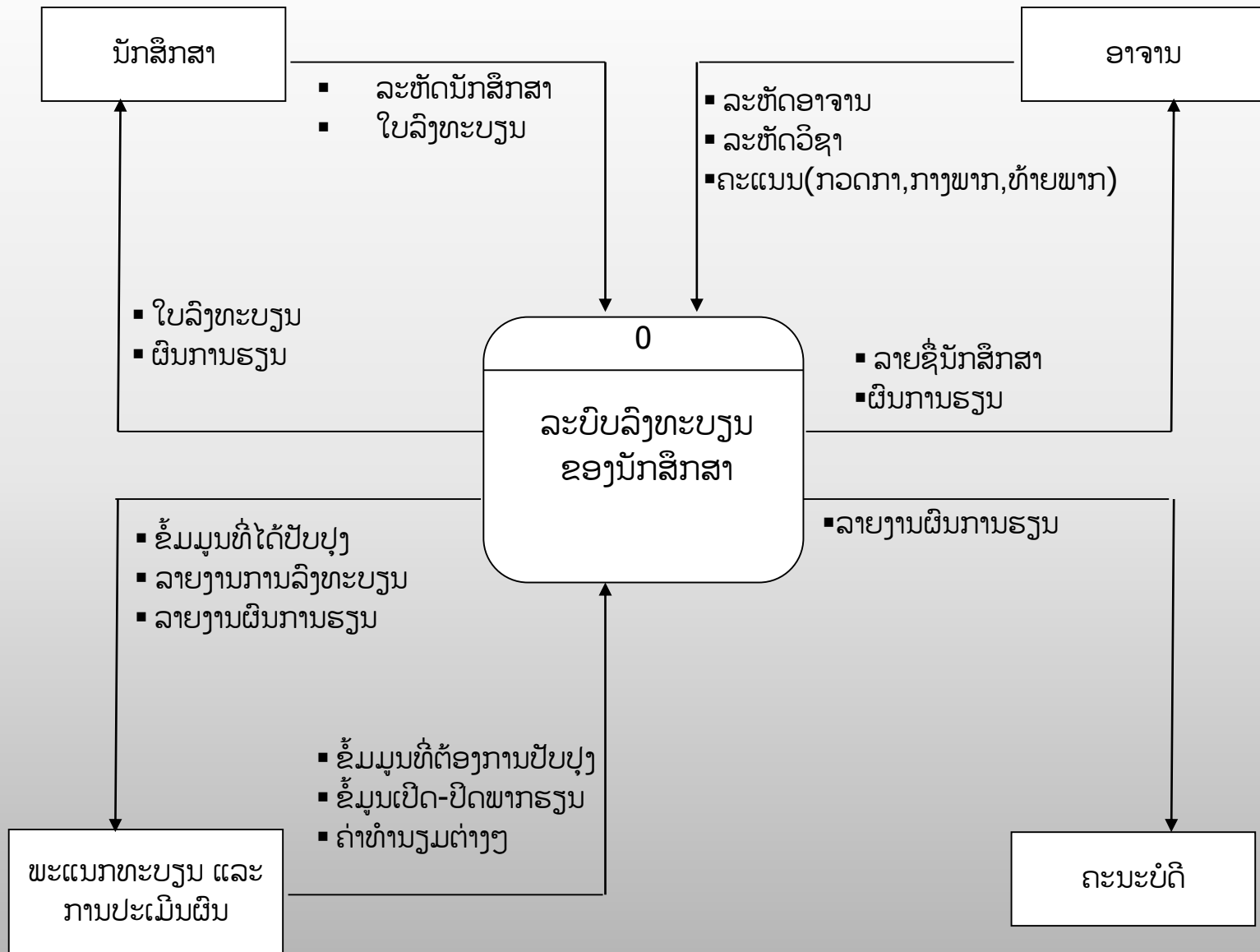
ແຜນວາດເນື້ອຫາເປັນແຜນວາດທີ່ສະແດງເຖິງພາບລວມການເຮັດວຽກຂອງລະບົບໂດຍມີ External Entities ແລະ ການໄຫຼຂໍ້ມູນທັງໝົດທີ່ໄຫຼເຂົ້າ-ອອກໃນລະບົບ ດ້ວຍການສະແດງໃນແຜນວາດດຽວ ໃນຂະນະດຽວກັນລະບົບທັງໝົດຈະຖືກນຳສະເໜີຢູ່ພາຍໃຕ້ Process ດຽວເທົ່ານັ້ນ.

External Entities ຈະຊ່ວຍກຳນົດຂອບເຂດຂອງລະບົບໃຫ້ມີຄວາມຊັດເຈນຂຶ້ນ ໂດຍຂອບເຂດຈະຖືກກຳນົດຂຶ້ນຈາກການນຳສະເໜີຜ່ານ Process ພຽງແຕ່ Process ດຽວພ້ອມກັບ External Entities ທີ່ກ່ຽວຂ້ອງ, ໂດຍ External Entities ຈະສົ່ງຂໍ້ມູນເຂົ້າອອກໃນລະບົບພາຍໃຕ້ຂອບເຂດຂອງລະບົບທີ່ຖືກກຳນົດຂຶ້ນ, ແຕ່ບໍ່ມີ Data store ເນື່ອງຈາກ Data store ຖືເປັນສ່ວນໜຶ່ງຂອງການດຳເນີນງານພາຍໃຕ້ຂະບວນການ, ແຕ່ມັນຈະລະບຸຢູ່ໃນແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 ເປັນຕົ້ນໄປ.

ຕົວຢ່າງ

Context Diagramລະບົບລົງທະບຽນຮຽນ ຂອງວິທະຍາໄລແຫ່ງໜຶ່ງ
ໂດຍມີ 4 ພາກສ່ວນທີ່ກ່ຽວຂ້ອງຄື:

1. ນັກສຶກສາ
2. ອາຈານ
3. ພະແນກທະບຽນ ແລະ ການປະເມີນຜົນ
4. ຄະນະບໍດີ



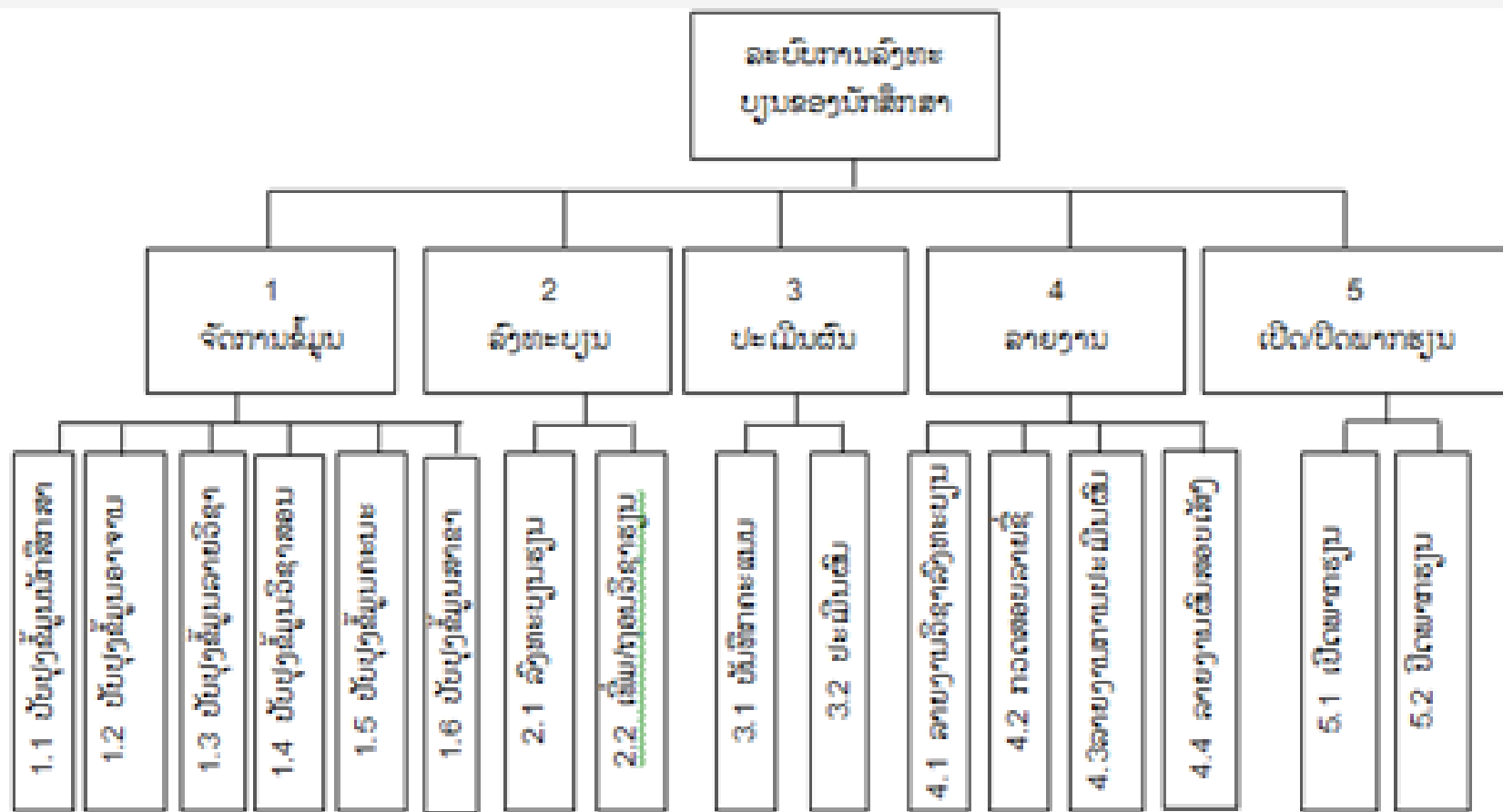
ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 (DFD Level1)

DFD Level1 ແມ່ນການນຳເອົາແຜນ Context Diagram ມາແຕກລາຍລະອຽດໂດຍຈະສະແດງເຖິງ Process ຫຼັກ ແລະ ສິ່ງທີ່ກ່ຽວຂ້ອງກັບລະບົບໂດຍລວມ ພ້ອມທັງ Data Store . ການຂຽນແຜນວາດຄວນຈະໃຫ້ເບິ່ງງ່າຍ ແລະ ໃຫ້ຢູ່ໃນໜ້າດຽວກັນ. ສ່ວນລູກສອນທີ່ສະແດງເຖິງການໄຫຼຂໍ້ມູນບໍ່ຄວນເຕັງກັນ ຫຼື ຕັດກັນ.

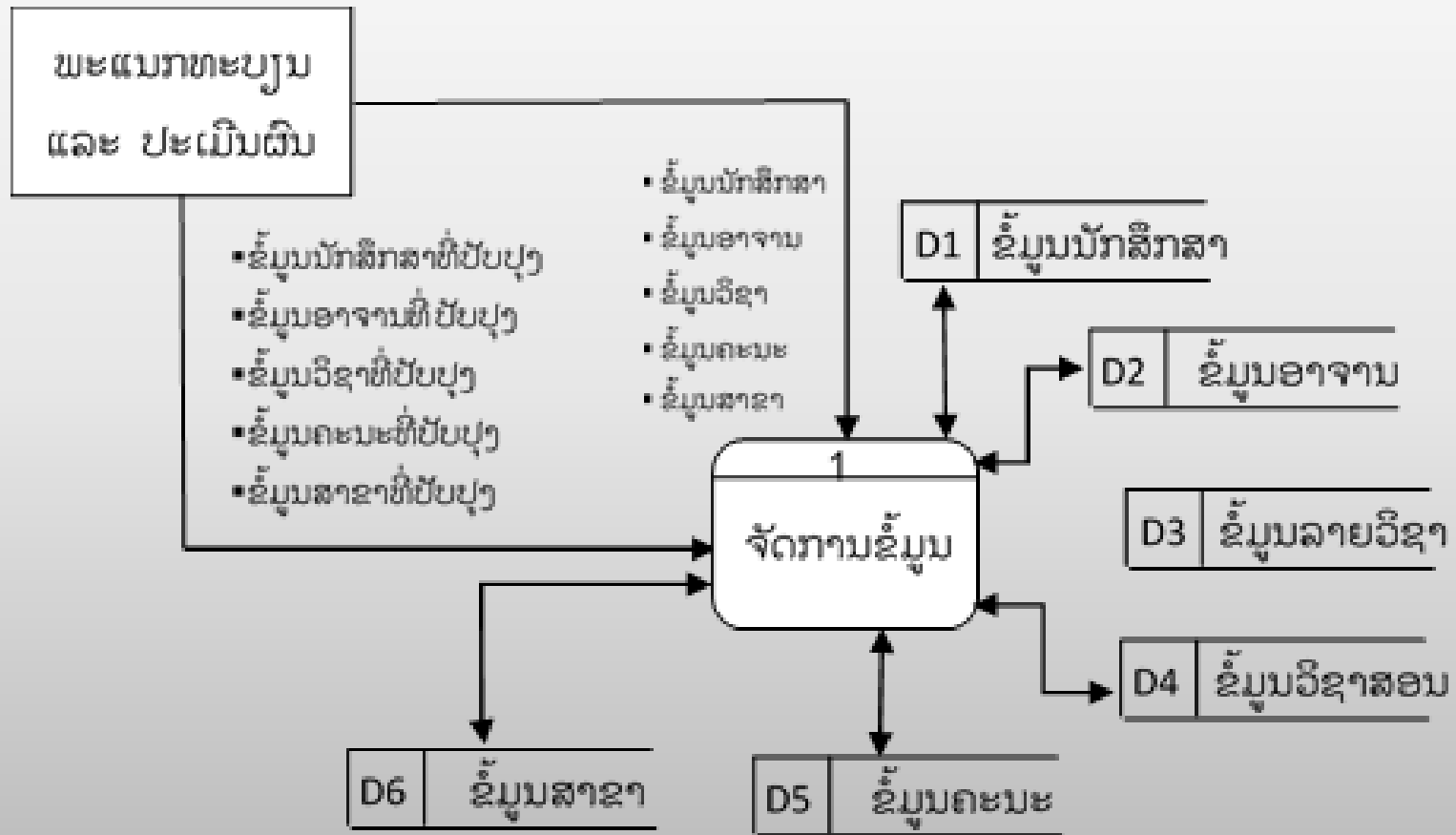
ຕົວຢ່າງ: ລະບົບລົງທະບຽນຮຽນແບບ Online

External Entity	Process	Data Store
ນັກສຶກສາ ອາຈານ ພະແນກທະບຽນ ແລະ ປະເມີນຜົນ ຄະນະບໍດີ	<ol style="list-style-type: none"> ຈັດການຂໍ້ມູນ <ol style="list-style-type: none"> ປັບປຸງຂໍ້ມູນນັກສຶກສາ ປັບປຸງຂໍ້ມູນອາຈານ ປັບປຸງຂໍ້ມູນລາຍວິຊາ ປັບປຸງຂໍ້ມູນວິຊາສອນ ປັບປຸງຂໍ້ມູນຄະນະ ປັບປຸງຂໍ້ມູນສາຂາ ລົງທະບຽນ <ol style="list-style-type: none"> ລົງທະບຽນ ເພີ່ມ/ຖອນວິຊາຮຽນ ປະເມີນຜົນ <ol style="list-style-type: none"> ບັນທຶກຄະແນນ ປະເມີນຜົນ ລາຍງານ <ol style="list-style-type: none"> ພິມລາຍງານການລົງທະບຽນແຕ່ລະວິຊາ ພິມໃບກວດສອບລາຍຊື່ ພິມລາຍງານການປະເມີນຜົນ ພິມລາຍງານການສອບເສັງ ເປີດ/ປິດພາກຮຽນ <ol style="list-style-type: none"> ປິດພາກຮຽນ ເປີດພາກຮຽນ 	D1 ຂໍ້ມູນນັກສຶກສາ D2 ຂໍ້ມູນລົງທະບຽນ D3 ຂໍ້ມູນລາຍວິຊາ D4 ຂໍ້ມູນອາຈານສອນ D5 ຂໍ້ມູນວິຊາສອນ D6 ຂໍ້ມູນຄະນະ D7 ຂໍ້ມູນສາຂາ D8 ຂໍ້ມູນພາກຮຽນ

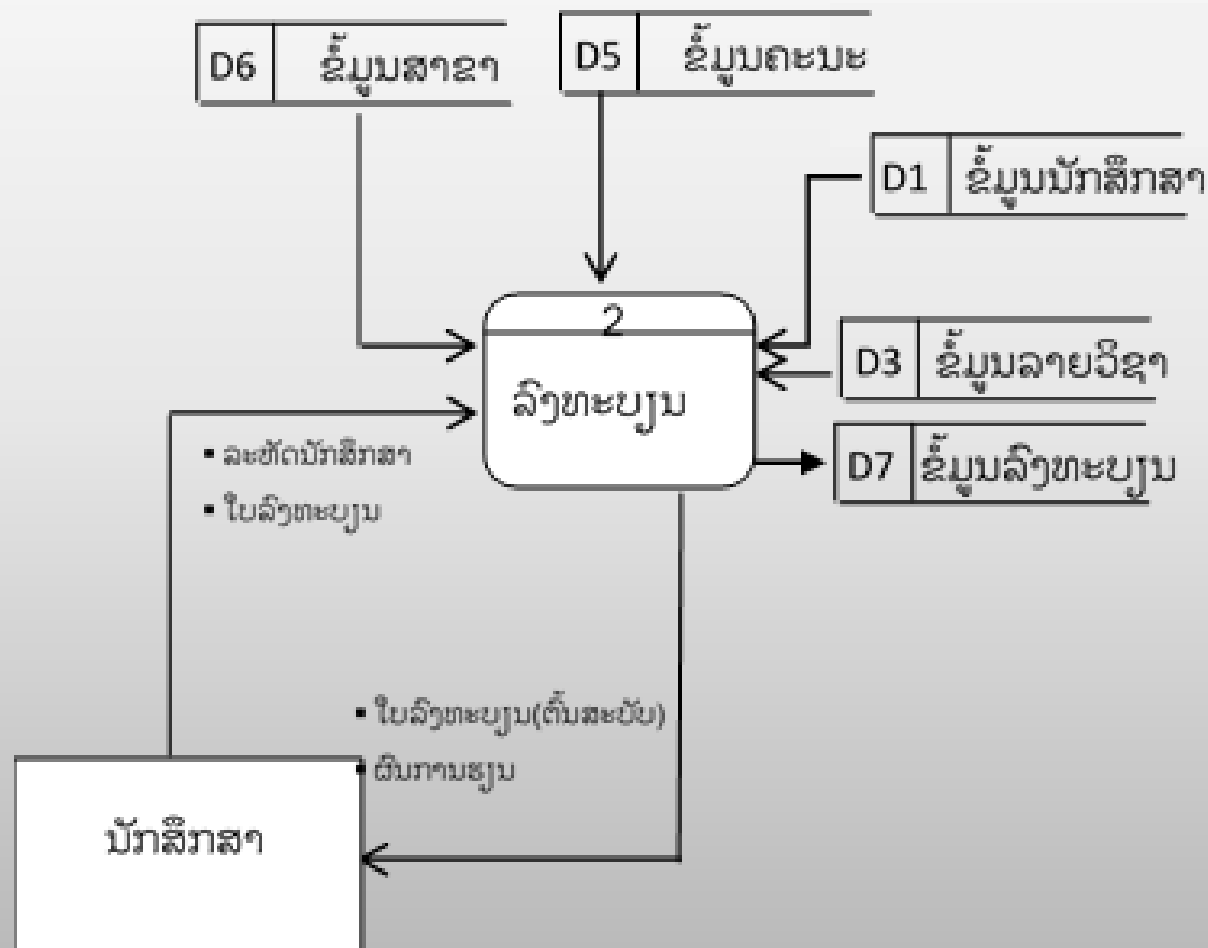
Functional Diagram



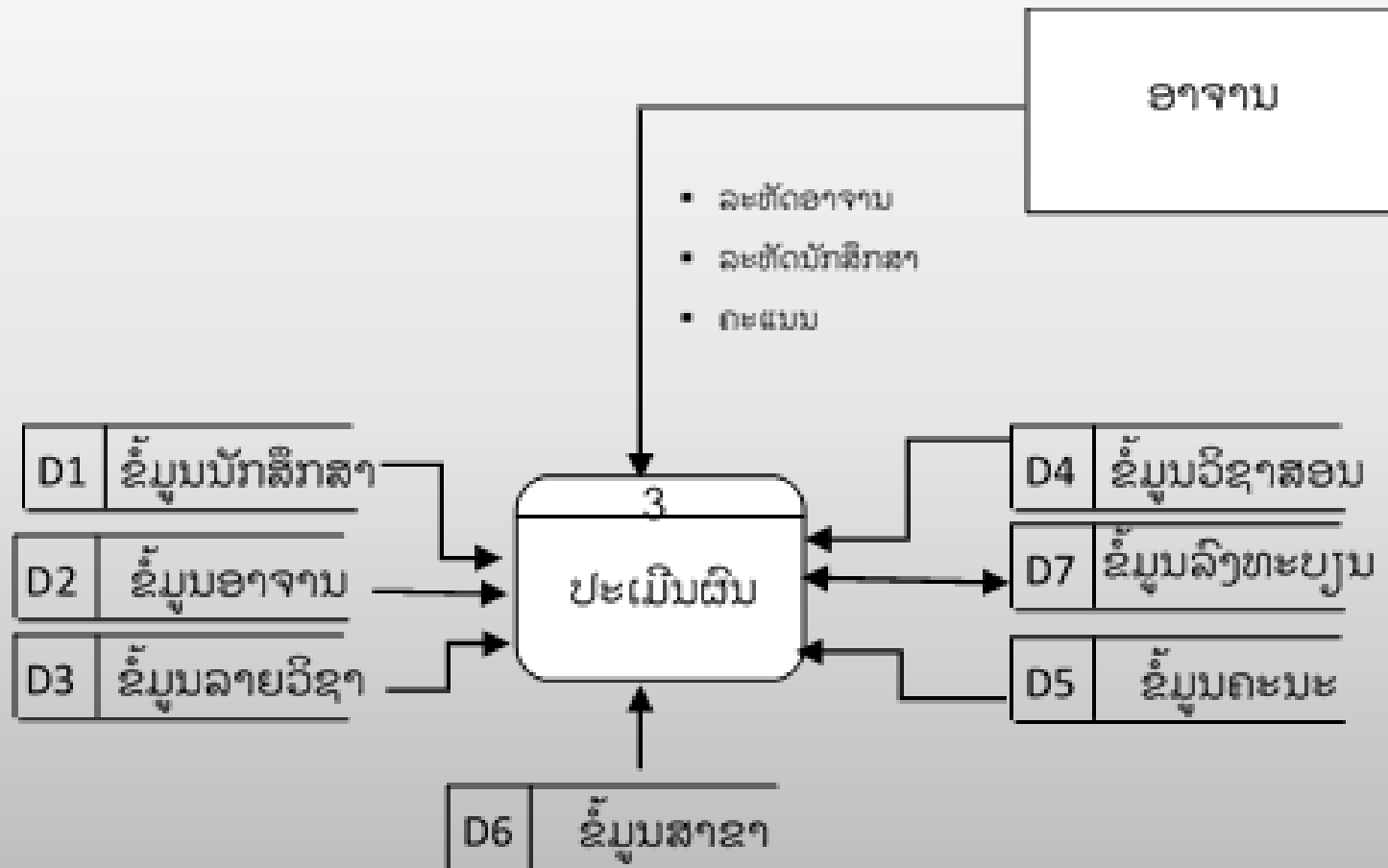
ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 Fragment 1



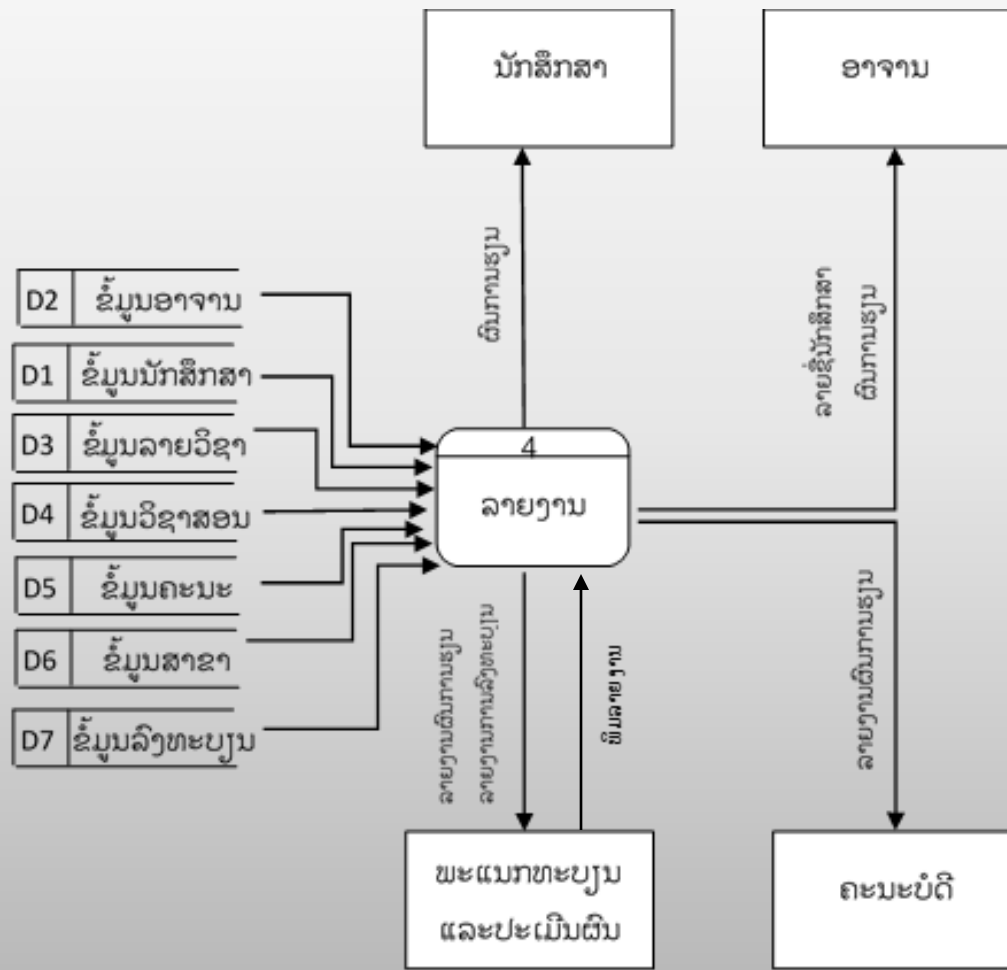
ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 Fragment 2



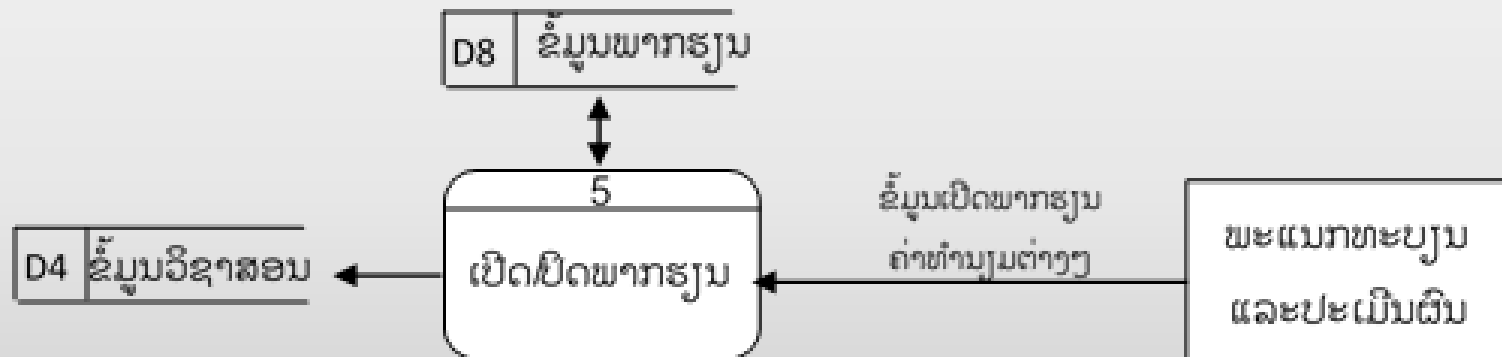
ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 Fragment 3



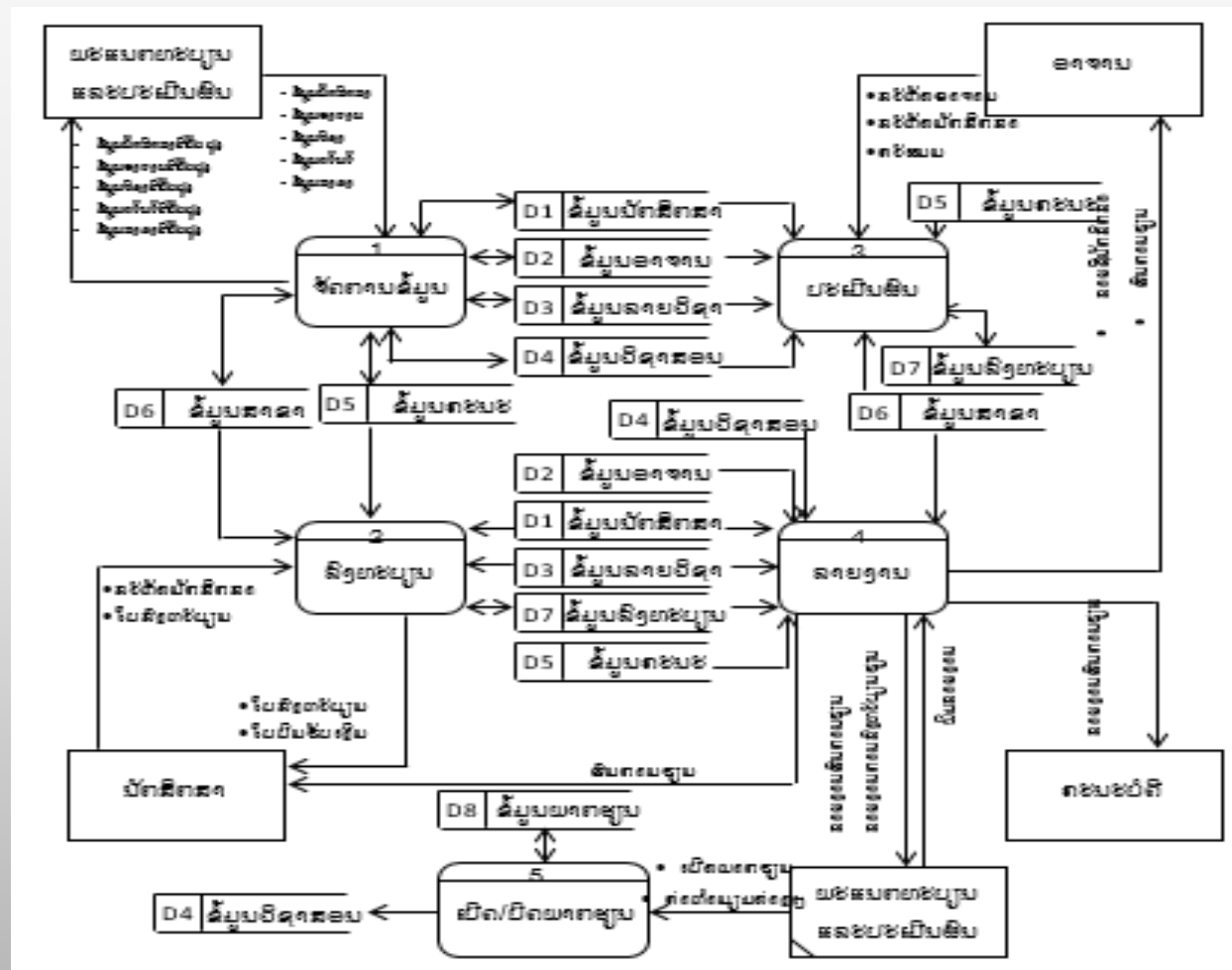
ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 Fragment 4



ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 Fragment 5



ສະຫຼຸບແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 ທັງ 5 ໜ້າວຽກ



ຕົວຢ່າງ: ການສ້າງແຜນວາດການໄຫຼຂໍ້ມູນ

ຫົວຂໍ້: ລະບົບບໍລິການເຊົ່າລົດ ຂອງບໍລິສັດ BM Car Rent Service Center

ລາຍລະອຽດຂອງບັນຫາ

ເນື່ອງຈາກຂະບວນການທຳງານທີ່ດຳເນີນຢູ່ໃນປະຈຸບັນຂອງລະບົບເຊົ້າລົດຍັງເປັນລະບົບການປະມວນຜົນດ້ວຍມືສ່ວຍຫຼາຍ, ເຖິງແມ່ນວ່າຈະມີການໃຊ້ຄອມພິວເຕີມາຊ່ວຍຈັດເກັບຂໍ້ມູນແຕ່ກໍບໍ່ເປັນລະບົບ, ລູກຄ້າ ທີ່ເຂົ້າມາໃຊ້ບໍລິການບາງຄັ້ງຕ້ອງໃຊ້ເວລາລໍຖ້າດົນເນື່ອງຈາກຕ້ອງການຄົ້ນຫາຂໍ້ມູນເກົ່າ ແລະ ຕ້ອງໃຊ້ເວລາ ໃນການກວດສອບຂໍ້ມູນລູກຄ້າ ເພື່ອປະກອບການເຮັດສັນຍາເຊົ້າ, ເຊິ່ງຍັງມີບາງຂັ້ນຕອນຊ້າຊ້ອນຢູ່ເຮັດໃຫ້ ລູກຄ້າບໍ່ໄດ້ຮັບຄວາມສະດວກໃນດ້ານການບໍລິການ, ລວມໄປເຖິງລະບົບເດີມບໍ່ສາມາດກວດສອບປະຫວັດລົດ ໃນດ້ານຂອງຂໍ້ມູນການສ້ອມແປງລົດເນື່ອງຈາກຂາດລະບົບການຈັດເກັບທີ່ດີພໍ. ສ່ວນໃນດ້ານຂອງການ ລາຍງານໃຫ້ຜູ້ບໍລິຫານມັກມີຄວາມຊັກຊ້າ, ບໍ່ກົງກັບເວລາ, ມີຂໍ້ຜິດພາດຢູ່ເລື້ອຍໆເຮັດໃຫ້ຂໍ້ມູນຂາດຄວາມ ໝ້າເຊື່ອຖື ດັ່ງນັ້ນ ຈຶ່ງເຫັນສົມຄວນພັດທະນາລະບົບໃໝ່ດ້ວຍການນຳເອົາລະບົບຂ່າວສານເຂົ້າມາໃຊ້ເພື່ອ ແກ້ໄຂບັນຫາດັ່ງກ່າວ.

ຈຸດປະສົງ

- ເພື່ອຫຼຸດຄວາມຊ້າຊ້ອນ ແລະ ຫຼຸດຂັ້ນຕອນການເຮັດວຽກໃຫ້ສັ້ນລົງ
- ເພື່ອເຮັດໃຫ້ການຈັດເກັບຂໍ້ມູນມີຄວາມຖືກຕ້ອງ, ເປັນລະບຽບ ແລະ ສາມາດຄົ້ນຫາໄດ້ຢ່າງວ່ອງໄວ
- ເພື່ອໃຫ້ລູກຄ້າໄດ້ຮັບການບໍລິການທີ່ດີ ມີຄວາມສະດວກ ແລະ ວ່ອງໄວ
- ເພື່ອໃຫ້ການຈັດພິມລາຍງານຕ່າງໆເປັນໄປແບບອັດຕະໂນມັດ ແລະ ສາມາດນຳສະເໜີຕໍ່ຜູ້ບໍລິຫານໄດ້ທັນເວລາ

ຂອບເຂດຂອງລະບົບ

- ວິເຄາະ ແລະ ອອກແບບລະບົບເຊົ່າລົດຂອງບໍລິສັດ BM Car Rent Service Center ໂດຍລະບົບທີ່ດໍາເນີນການຈະເປັນຮູບແບບຂອງລະບົບເຄືອຄ່າຍທ້ອງຖິ່ນ ແລະ ຍັງສາມາດເຊື່ອມໂຍງເຂົ້າກັບເຄືອຄ່າຍອື່ນເຕີເນັດໄດ້.
- ລະບົບຂ່າວສານທີ່ພັດທະນາຂຶ້ນມາຈະຄວບຄຸມໄປເຖິງ ລະບົບເຊົ່າລົດ, ການຈອງລົດ, ການສົ່ງລົ ແລະ ການສ້ອມແປງລົດ

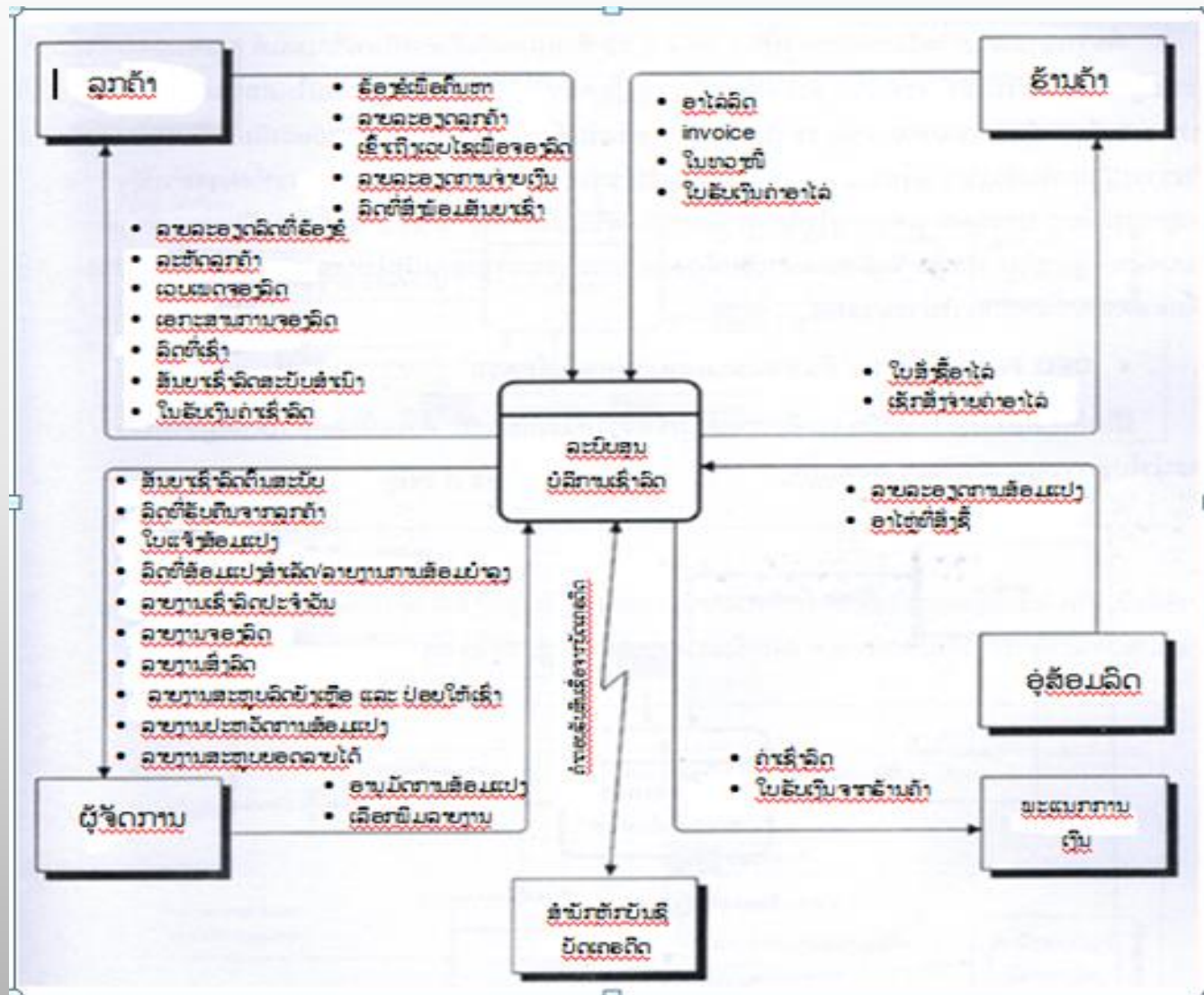
ຜົນປະໂຫຍດຄາດວ່າຈະໄດ້ຮັບ

- ລະບົບເຊົ່າລົດຈະເຮັດວຽກເປັນລະບົບຍິ່ງຂຶ້ນ
- ສ້າງຄວາມເຊື່ອໝັ້ນໃຫ້ກັບລູກຄ້າ

ລາຍລະອຽດຂອງລະບົບສູນບໍລິການເຊົ່າລົດທີ່ຈະນຳໄປປະກອບການສ້າງແຜນວາດການໄຫຼຂໍ້ມູນ

External Entity (ພາກສ່ວນທີ່ກ່ຽວຂ້ອງ)	Process (ໜ້າວຽກ)	Data Store (ຂໍ້ມູນ)
<ol style="list-style-type: none"> 1. ລູກຄ້າ 2. ຜູ້ຈັດການ 3. ພະແນກການເງິນ 4. ຮ້ານຄ້າ 5. ອຸສະຫມາດ 6. ສຳນັກຫັກບັນຊີບັດເຄຣດິດ 	<ol style="list-style-type: none"> 1. ຄົ້ນຫາ ແລະ ສະແດງລາຍລະອຽດລົດ <ol style="list-style-type: none"> 1.1 ຄົ້ນຫາລົດ 1.2 ສະແດງລາຍລະອຽດລົດ 2. ເຊົ່າລົດ <ol style="list-style-type: none"> 2.1 ກວດສອບ/ບັນທຶກປະຫວັດລູກຄ້າ 2.2 ດຶງຂໍ້ມູນລູກຄ້າ 2.3 ບັນທຶກລາຍການເຊົ່າ 2.4 ຈ່າຍຄ່າມັດຈຳ 2.5 ຍືນຍັນການເຊົ່າລົດ 3. ຈອງລົດ <ol style="list-style-type: none"> 3.1 load ເວບໄຊ 3.2 ບັນທຶກຂໍ້ມູນລູກຄ້າເບື້ອງຕົ້ນ 3.3 ບັນທຶກລາຍການຈອງລົດ 3.4 ຄິດໄລ່/ຈ່າຍເງິນຄ່າຈອງລົດ 3.5 ຍືນຍັນການຈອງ 4. ຮັບລົດຄືນ <ol style="list-style-type: none"> 4.1 ກວດສອບວັນສິ່ງຄືນລົດ 4.2 ຄິດໄລ່ຄ່າເຊົ່າລົດ 4.3 ຈ່າຍເງິນຄ່າເຊົ່າລົດ 4.4 ຍືນຍັນການຈ່າຍເງິນ 5. ສ້ອມແປງລົດ <ol style="list-style-type: none"> 5.1 ພິມໃບແຈ້ງສ້ອມແປງ 5.2 ຈັດຊື້ອາໄລ 5.3 ຈ່າຍເງິນຄ່າອາໄລ 5.4 ສ້ອມແປງລົດ 5.5 ສົ່ງມອບລົດທີ່ສ້ອມແປງສຳເລັດ 6. ພິມລາຍງານ 	<ol style="list-style-type: none"> 1. ຂໍ້ມູນລູກຄ້າ 2. ຂໍ້ມູນລົດ 3. ຂໍ້ມູນສັນຍາເຊົ່າ 4. ຂໍ້ມູນລາຍການເຊົ່າ 5. ຂໍ້ມູນລາຍການຈອງລົດ 6. ຂໍ້ມູນໃບສັ່ງສ້ອມແປງ 7. ຂໍ້ມູນການສ້ອມແປງລົດ

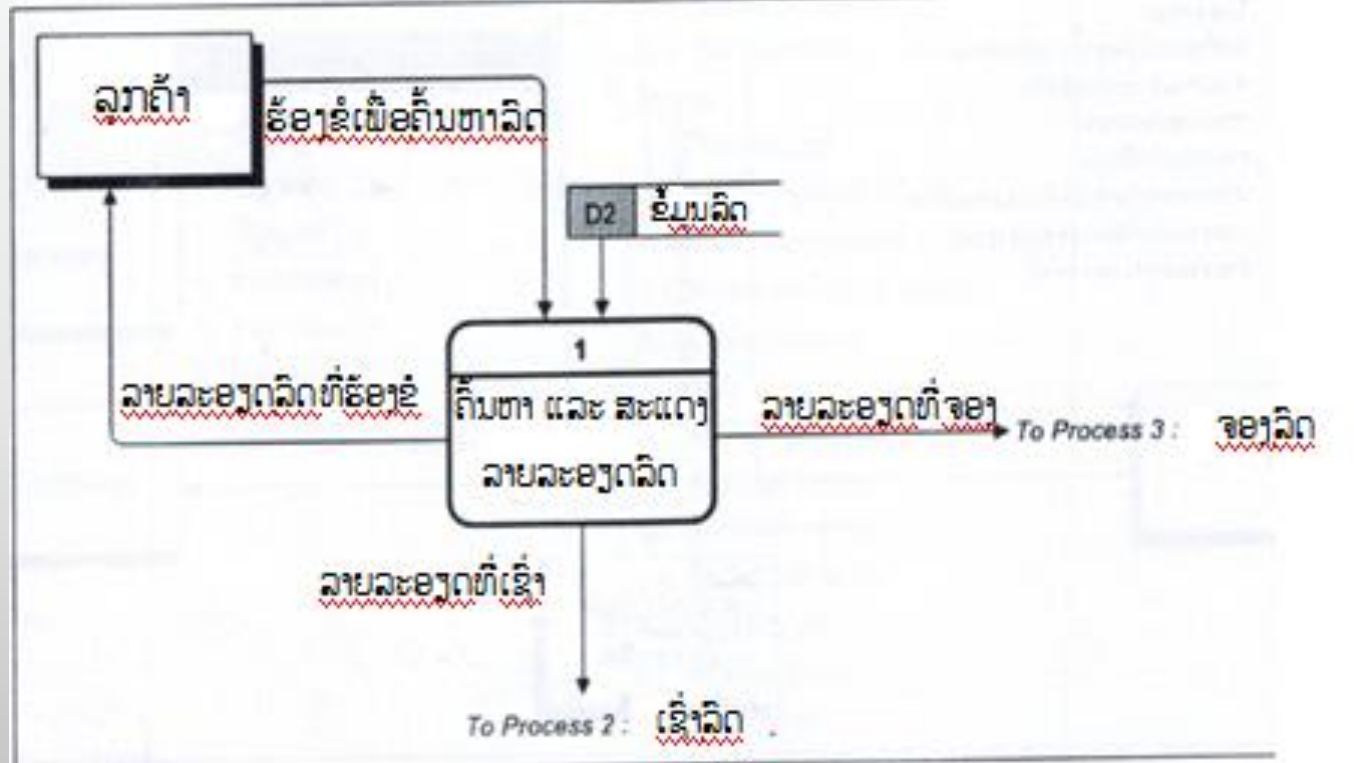
Context Diagram ຂອງລະບົບສູນບໍລິການເຊົ່າລົດ



DFD Fragments

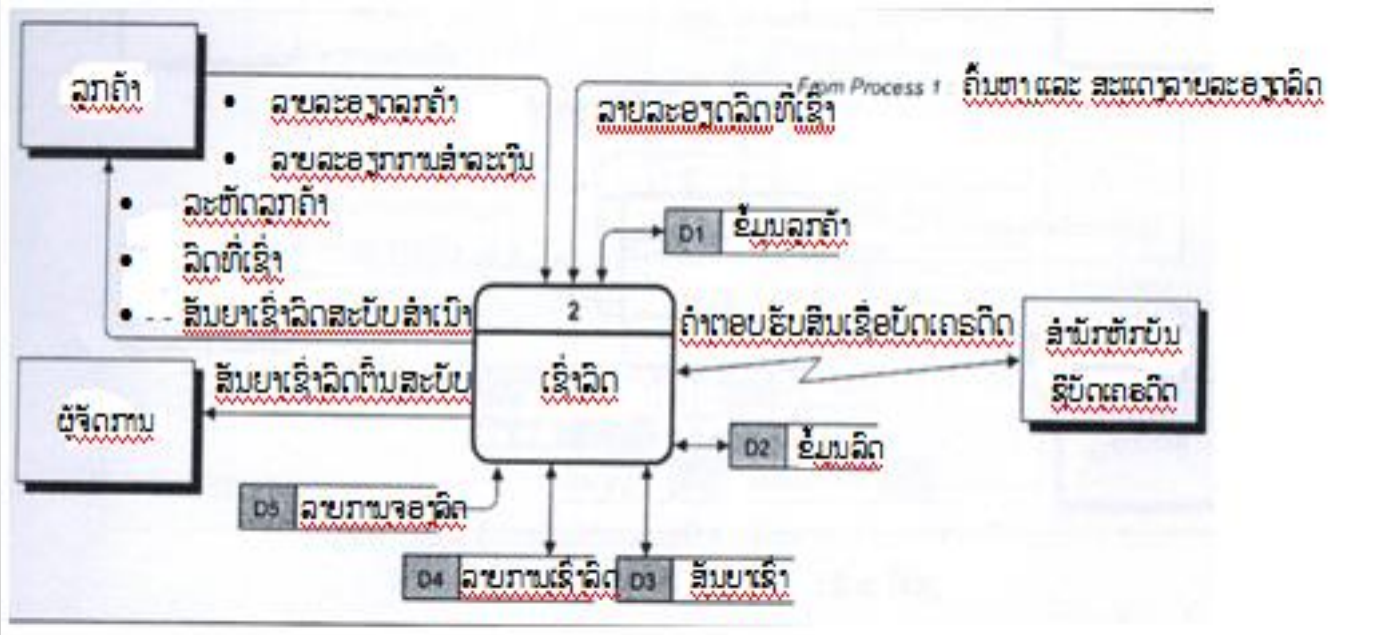
ໃນແຜນວາດການໄຫຼຂໍ້ມູນໃນລະບົບສູນບໍລິການເຊົ່າລົດຈະ
ມີທັງໝົດຢູ່ 6 Fragment. ຈະສັງເກດເຫັນວ່າໃນແຕ່ລະ
Fragment ໄດ້ສະແດງສະຖານນະການທີ່ກ່ຽວຂ້ອງກັບ
Process ຫຼັກໜຶ່ງ Process ເຊັ່ນ: ເຊົ່າລົດ, ຈອງລົດ ຫຼື ສົ່ງລົດ
ເປັນຕົ້ນ ໂດຍລາຍລະອຽດທີ່ສະແດງຈະກ່ຽວຂ້ອງກັບການປະຕິສໍາ
ພັນກັນລະຫວ່າງ Process, External Entity ແລະ Data
Store

DFD Fragment 1: ຄົ້ນຫາ ແລະ ສະແດງລາຍລະອຽດລົດ



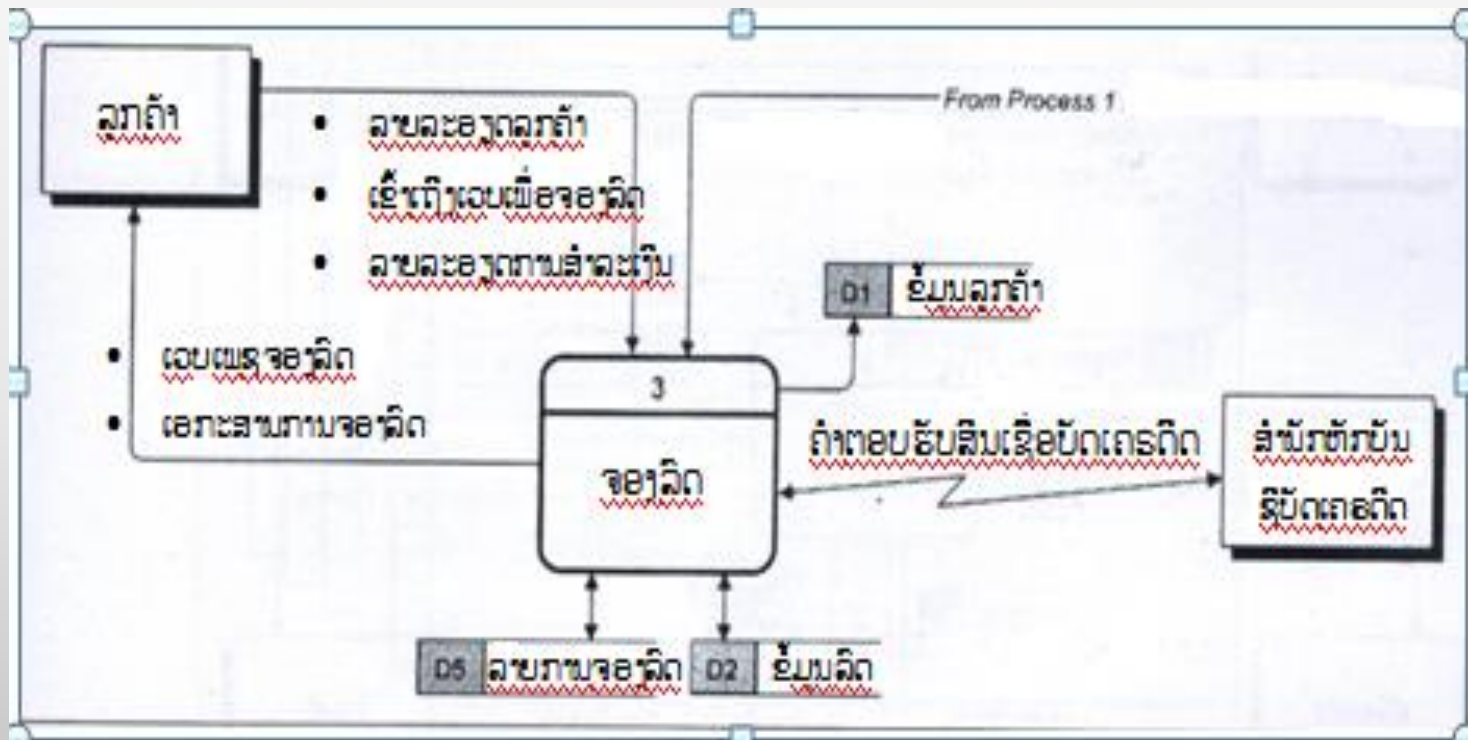
ເປັນ DFD ຄົ້ນຫາ ແລະ ສະແດງລາຍລະອຽດຂອງລົດ. ເມື່ອລູກຄ້າໄດ້ຂໍ້ມູນລົດທີ່ຕ້ອງການແລ້ວກໍຈະນຳໄປສູ່ການຈອງລົດ ຫຼື ເຊົ່າລົດຕໍ່ໄປ

DFD Fragment 2: ເຊົ່າລົດ



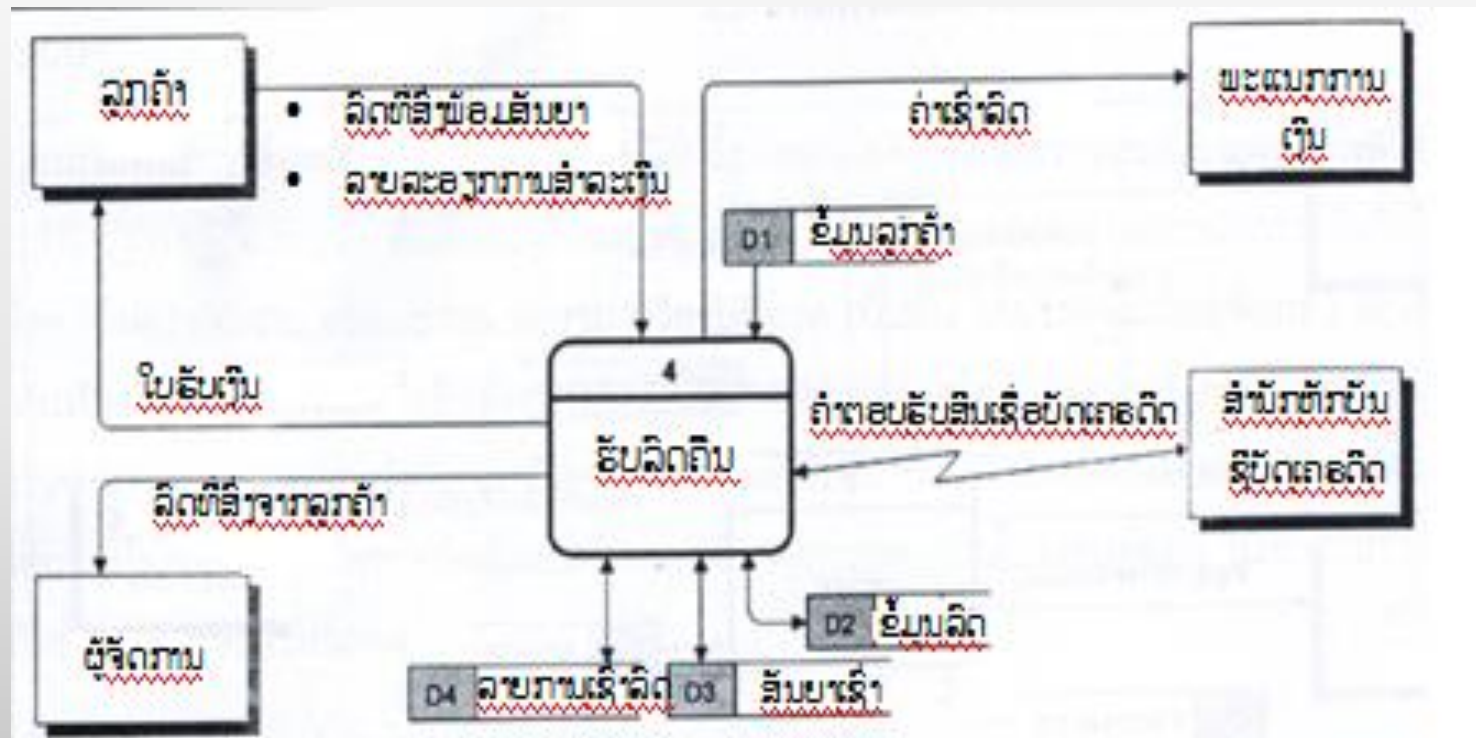
ເມື່ອລູກຄ້າໄດ້ຂໍ້ມູນທີ່ຕ້ອງການເຊົ່າແລ້ວ, ລູກຄ້າຈະຕ້ອງໃຫ້ລາຍລະອຽດຂອງຕົນ ເພື່ອນຳໄປໃຊ້ປະກອບການເຮັດສັນຍາເຊົ່າລົດ ພ້ອມຄ່າມັດຈຳ ແລະ ເມື່ອດຳເນີນການ ສໍາເລັດແລ້ວກໍຈະໄດ້ລົດທີ່ເຊົ່າໄປ

DFD Fragment 3: ຈອງລົດ



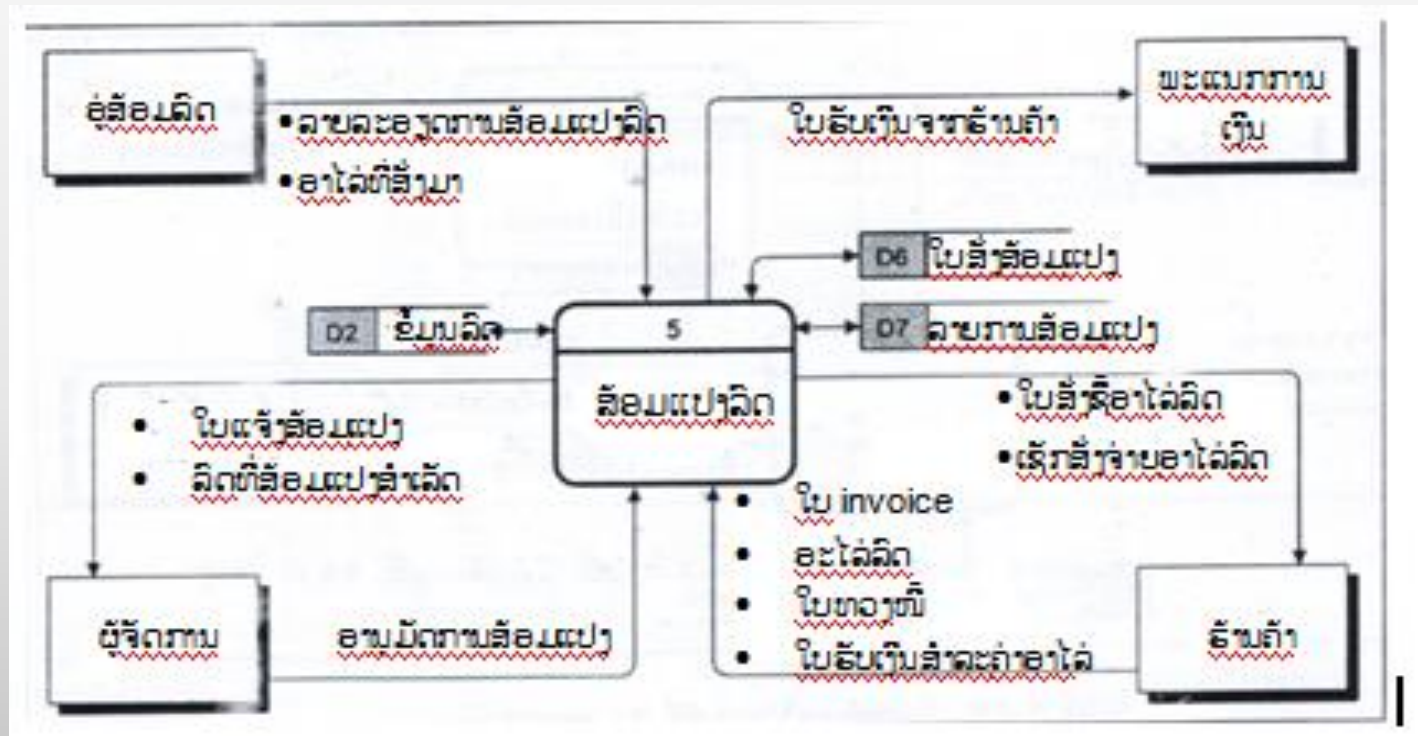
ເມື່ອລູກຄ້າສາມາດຈອງລົດຜ່ານເວບໄຊດ້ວຍຕົນເອງໄດ້ ແລະ ເມື່ອໄດ້ລາຍລະອຽດທີ່ຕ້ອງການຈອງແລ້ວລູກຄ້າຈະຕ້ອງຈ່າຍເງິນຜ່ານບັດເຄຣດິດເພື່ອຢືນຢັນການຈອງລົດດັ່ງກ່າວ

DFD Fragment 4 : ຮັບລົດຄືນ



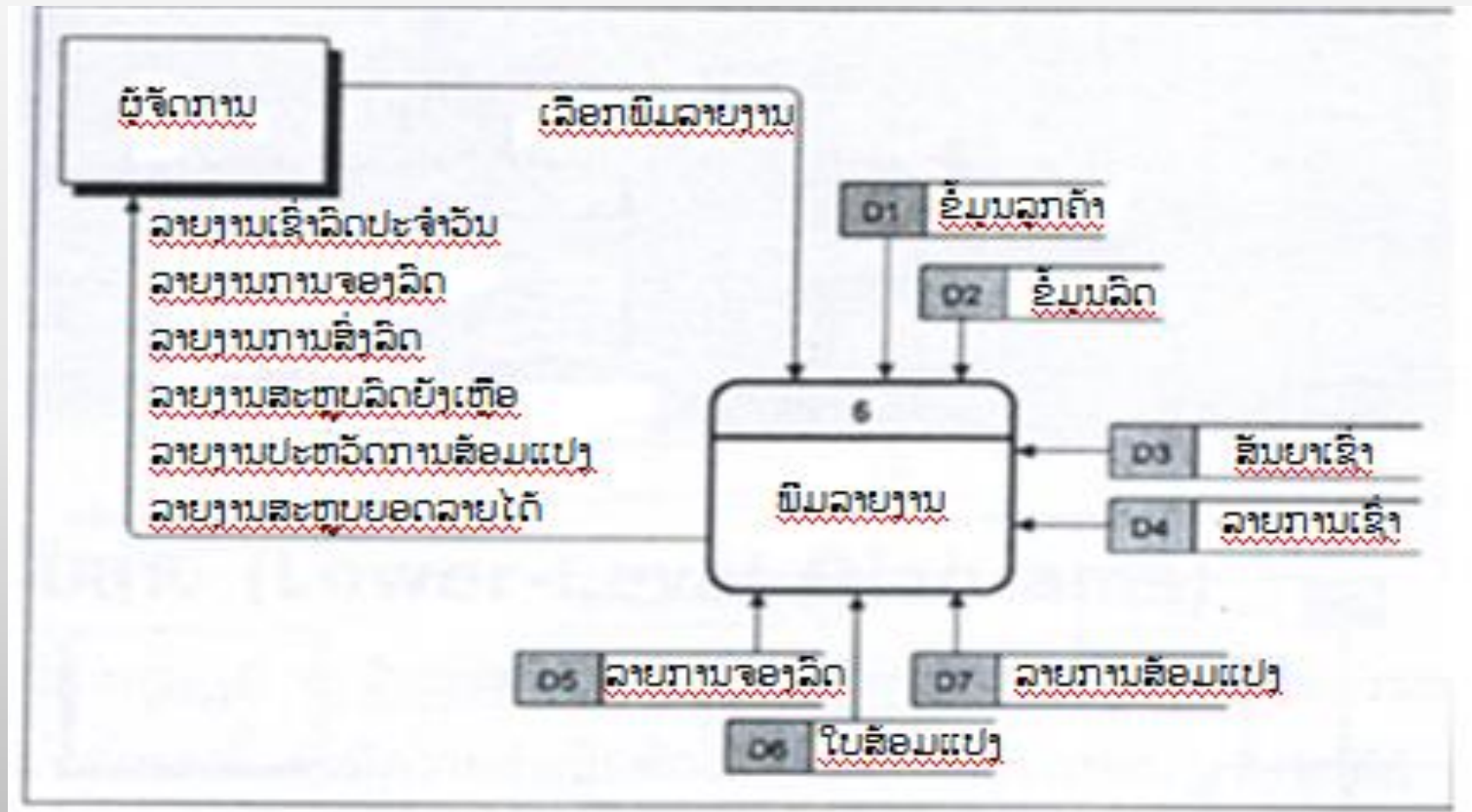
ເມື່ອລູກຄ້າເຊົາລົດຄົບກໍານົດແລ້ວ ລູກຄ້າຈະຕ້ອງສົ່ງລົດພ້ອມທັງສໍາລະເງິນທີ່ຍັງເຫຼືອ, ແຕ່ ຖ້າສິ່ງກາຍກໍານົດຈະຕ້ອງໄດ້ເສັຍຄ່າປັບໄໝ.

DFD Fragment 5 : ສ້ອມແປງລົດ



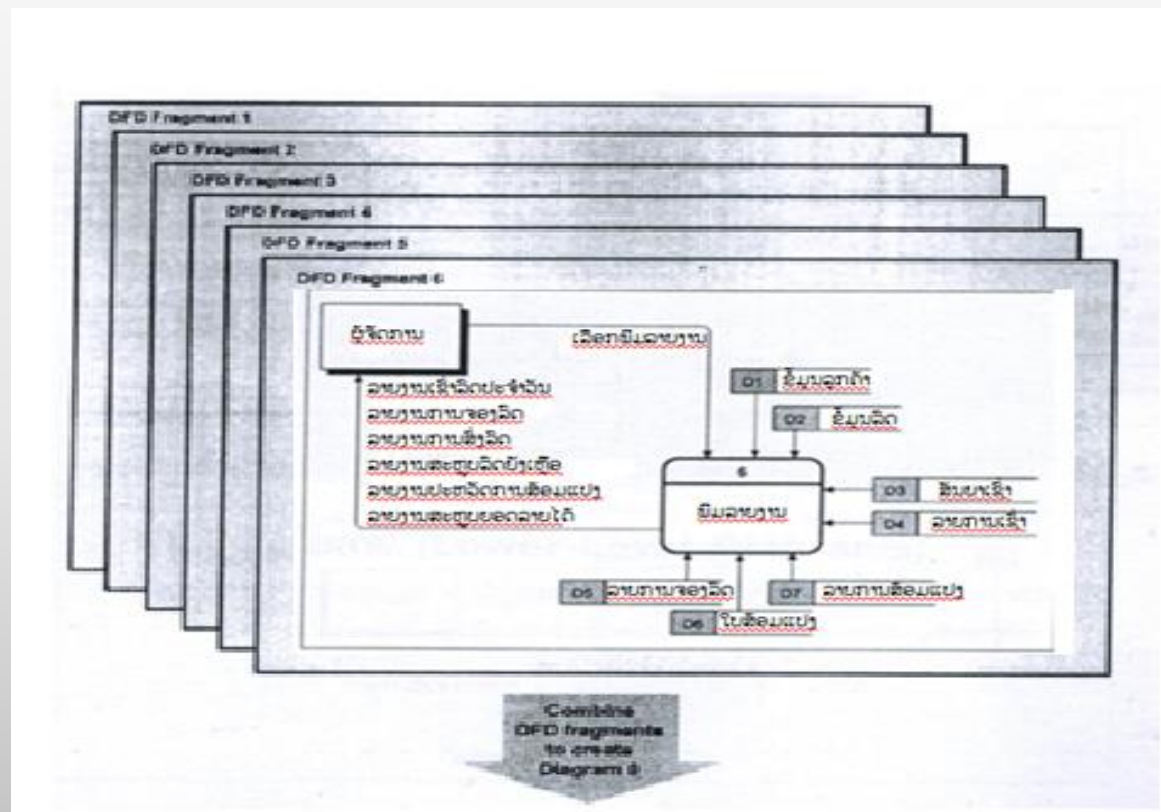
ອຸ່ສ້ອມແປງລົດຈະກຳນົດລາຍລະອຽດການສ້ອມແປງເພື່ອນຳໄປເຮັດໃບແຈ້ງສ້ອມແປງ ໃຫ້ຜູ້ຈັດການອານຸມັດ, ເມື່ອມີການສັງເກດໄດ້ລົດ ແລະ ຮ້ານຄ້າໄດ້ຈັດສົ່ງສິນສ່ວນອາ ໄຫຼມາໃຫ້ກໍຈະດຳເນີນການສ້ອມແປງໃຫ້ສໍາເລັດ, ໃນຂະນະດຽວກັນທາງບໍລິສັດກໍຈະສໍ ລະເງິນຄ່າອາໄຫຼແກ່ທາງຮ້ານຜ່ານເຊັກສັງຈ່າຍ

DFD Fragment 6 : ພິມລາຍງານ



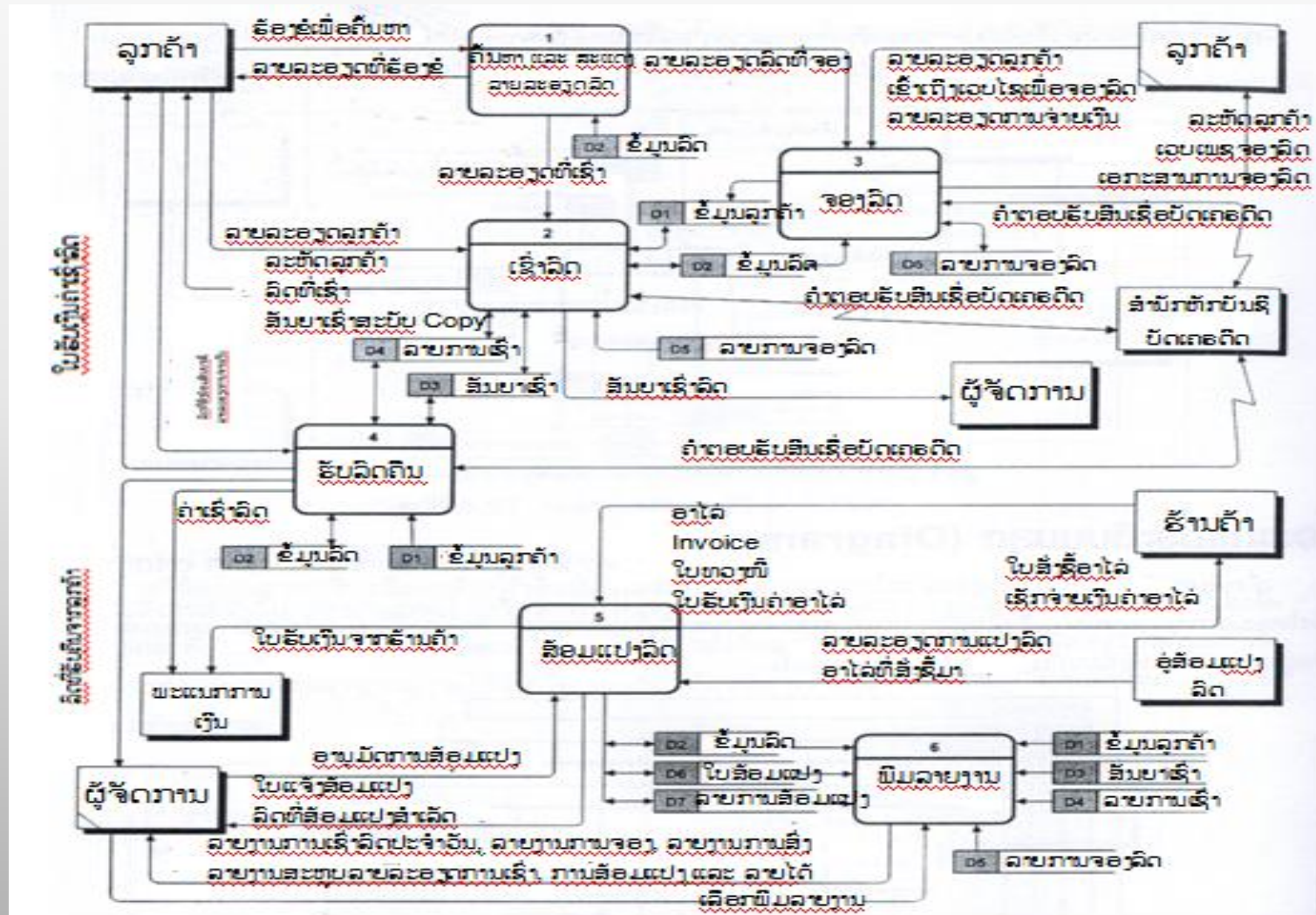
ເປັນ DFD ພິມລາຍງານຕ່າງໆສິ່ງໃຫ້ແກ່ຜູ້ຈັດການ

Diagram 0



ການນຳເອົາ DFD Fragment ທັງ 6 ມາລວມເຂົ້າກັນເພື່ອສ້າງເປັນ Diagram ລະດັບ 0 ຂອງລະບົບບໍລິການເຊົ່າລົດ

DFD Level 1 ຂອງລະບົບເຊົ່າລົດ



ຄຳອະທິບາຍການປະມວນຜົນຂອງຂໍ້ມູນ

ຄຳອະທິບາຍການປະມວນຜົນຈະຊ່ວຍໃຫ້ເຫັນໄດ້ຂັ້ນຕອນການເຮັດວຽກຢ່າງລະອຽດຂອງແຕ່ລະ Process ໃນຮູບແບບ Algorithm ໂດຍຈຳນວນຂອງຄຳອະທິບາຍການປະມວນຜົນຂໍ້ມູນຈະມີຫຼາຍໜ້ອຍພຽງໃດຂຶ້ນຢູ່ກັບຈຳນວນລາຍລະອຽດທີ່ໃຊ້ວຽກໃນແຕ່ລະລະດັບຂອງແຜນວາດການໄຫຼຂໍ້ມູນ.

ຈຸດປະສົງຂອງການຂຽນຄໍາອະທິບາຍ

- ເພື່ອຫຼຸດຄວາມບໍ່ຈະແຈ້ງຂອງ Process ແລະ ໃຊ້ເປັນກົດຂອງນັກວິເຄາະເພື່ອຮຽນຮູ້ລາຍລະອຽດທີ່ຖືກຕ້ອງຂອງແຕ່ລະ Process
- ເພື່ອຄວາມທ່ຽງຕົງ ເຊິ່ງຂໍ້ກຳນົດທີ່ໄດ້ກຳນົດໄວ້ໃນຄໍາອະທິບາຍນັກຂຽນໂປຣແກຣມສາມາດນຳໄປອອກແບບ ແລະ ພັດທະນາໂປຣແກຣມໄດ້ ດັ່ງນັ້ນ ຖ້າຄໍາອະທິບາຍບໍ່ຊັດເຈນກໍຈະເຮັດໃຫ້ເກີດຄວາມເຂົ້າໃຈບໍ່ກົງກັນ
- ເພື່ອໃຊ້ກວດສອບໃນຂັ້ນຕອນການອອກແບບລະບົບເພື່ອໃຫ້ເກີດຄວາມໝັ້ນໃຈ process ທີ່ໄດ້ຮັບ input ເຂົ້າມາປະມວນຜົນຈະໄດ້ຜົນຮັບຕາມທີ່ໄດ້ກຳນົດໄວ້ໃນແຜນວາດການໄຫຼຂໍ້ມູນ

ຕົວຢ່າງ: ຄໍາອະທິບາຍການປະເມີນຜົນຂອງລະບົບ ລົງທະບຽນຮຽນ

System:	ລະບົບການລົງທະບຽນຂອງນັກສຶກສາ
Process:	2, ລົງທະບຽນ
Date:	dd/mm/yyyy
Tasks or Activities:	2.1 ລົງທະບຽນຮຽນ 2.2 ເພີ່ມ/ຖອນ ວິຊາຮຽນ

System:	ລະບົບການລົງທະບຽນຂອງນັກສຶກສາ
Process:	2, ລົງທະບຽນ
Date:	dd/mm/yyyy
Tasks or Activities:	

ເປັນການລົງທະບຽນລາຍວິຊາຮຽນເຊິ່ງນັກສຶກສາສາມາດລົງທະບຽນ
ຮຽນໄດ້ສູງສຸດບໍ່ເກີນ 21 ໜ່ວຍກົດສຳລັບພາກການສຶກສາປົກກະຕິ ແລະ ລົງທະ
ບຽນຮຽນສູງສຸດບໍ່ເກີນ 9 ໜ່ວຍກົດສຳລັບພາກຮຽນສາມ

ປະເພດຂອງຄໍາອະທິບາຍການປະເມີນຜົນ

- Structured English
- Decision Tree
- Data Modeling

Structured English

ເປັນໂຄງສ້າງພາສາທີ່ມີລັກສະນະຄ້າຍຄື Algorithm ໂດຍ
ຈະມີຄໍາສັ່ງເປັນພາສາອັງກິດເຊິ່ງເອີ້ນ ວ່າຄໍາສັ່ງສະເພາະເຊັ່ນ :

BEGIN, REPEAT, IF, END, UNTIL, THEN,
CASE, WHILE, ELSE, OF, DO, FOR.

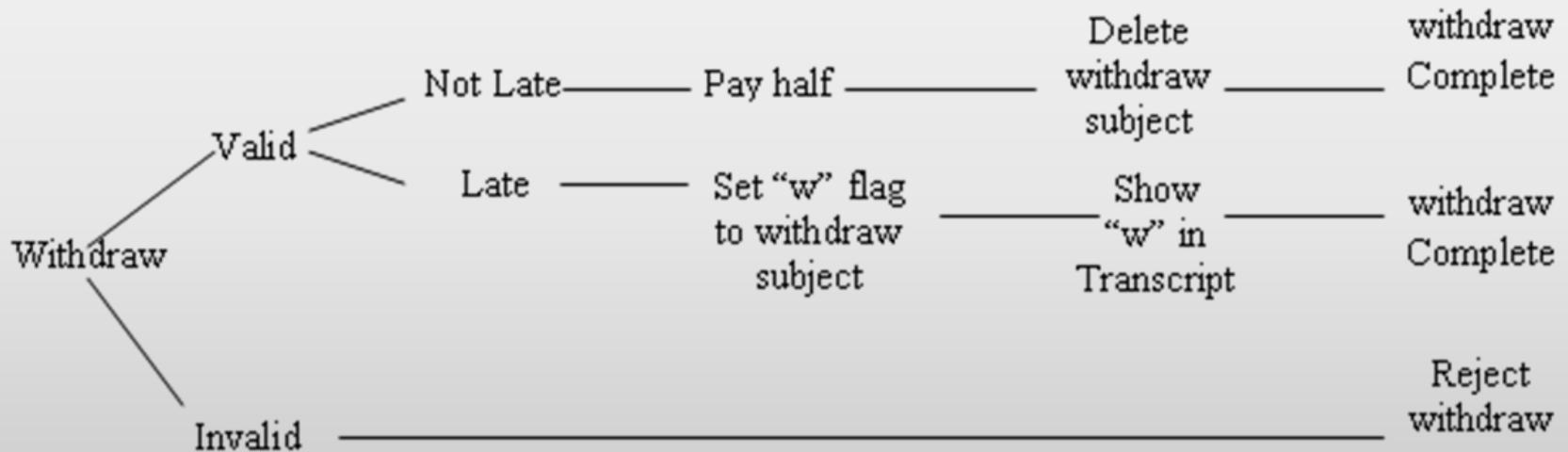
ຕົວຢ່າງ: ຄໍາອະທິບາຍ ລົງທະບຽນ ແບບ Structured English

```
IF normal semester
    THEN
        Max_Credit = 21
ELSE
    IF summer_semester
        THEN
            Max_Credit = 9
            .
            .
            .
    IF cum_credit > Max_Credit
        THEN
            Ignore_register
    ELSE
        Submit_Register
        Cumulative credit to cum_credit
        .
        .
        .
```

Decision Tree

ມີລັກສະນະເປັນໂຄງສ້າງເງື່ອນໄຂການຕັດສິນໃຈທີ່ມີ
ລັກສະນະຄືຕົ້ນໄມ້ທີ່ແບ່ງແຍກເງື່ອນໄຂ ອອກເປັນສ່ວນໆເຮັດໃຫ້
ເຫັນໂຄງສ້າງໄດ້ຊັດເຈນຂຶ້ນ. ດັ່ງນັ້ນ ຖ້າ Process ໃດທີ່ມີ
ເງື່ອນໄຂຊັບຊ້ອນຫຼາຍ ການນຳເອົາ Decision Tree ມາໃຊ້ເພື່ອ
ອະທິບາຍການປະມວນຜົນກໍຖືວ່າເປັນແນວທາງໜຶ່ງທີ່ເໝາະສົມ.

ຕົວຢ່າງ: ຄໍາອະທິບາຍຖອນວິຊາຮຽນ ແບບ Decision Tree



ຕົວຢ່າງ: ການຂຽນຄໍາອະທິບາຍການປະມວນຜົນ

ລະບົບບໍລິການເຊົ່າລົດ ຂອງບໍລິສັດ BM Car
Rent Service Center

Process 1: ຄົ້ນຫາ ແລະ ສະແດງລາຍລະອຽດລົດ

System DFD Number Process Name	ລະບົບບໍລິການເຊົ່າລົດ 1 ຄົ້ນຫາ ແລະ ສະແດງລາຍລະອຽດລົດ
Input Data Flows	ຮ້ອງຂໍເພື່ອຄົ້ນຫາລົດ
Output Data Flows	ລາຍລະອຽດລົດທີ່ຮ້ອງຂໍ, ລາຍລະອຽດລົດທີ່ເຊົ່າ, ລາຍລະອຽດລົດທີ່ຈອງ
Data Stores used	ຂໍ້ມູນລົດ
Description	Process ຫຼັກທີ່ນຳມາໃຊ້ເພື່ອການຄົ້ນຫາ ແລະ ສະແດງລາຍລະອຽດລົດ
Method	1.1 ຄົ້ນຫາລົດ 1.2 ສະແດງລາຍລະອຽດລົດ

Process 1.1

System DFD Number Process Name	ລະບົບບໍລິການເຊົ່າລົດ 1.1 ຄົ້ນຫາລົດ
Input Data Flows	ຮ້ອງຂໍເພື່ອຄົ້ນຫາລົດ
Output Data Flows	ລາຍລະອຽດລົດ
Data Stores used	ຂໍ້ມູນລົດ
Description	ຄົ້ນຫາລົດທີ່ຕ້ອງການໃຫ້ກັບລູກຄ້າເພື່ອສິ່ງລາຍລະອຽດໄປທີ່ຂັ້ນຕອນການຈອງ ຫຼື ເຊົ່າລົດ
Method	ລະບົບຈະກຽມກາຕາລັອກແບບ online ໃຫ້ກັບລູກຄ້າ. ຖ້າລູກຄ້າເຂົ້າມາເຊົ່າລົດໂດຍກົງກໍສາມາດຄົ້ນຫາຂໍ້ມູນລົດໄດ້ກໍສາມາດຄົ້ນຫາໄດ້ຈາກກາຕາລັອກ

Process 1.2

System DFD Number Process Name	ລະບົບບໍລິການເຊົ່າລົດ 1.2 ສະແດງລາຍລະອຽດລົດ
Input Data Flows	ລາຍລະອຽດລົດ
Output Data Flows	ລາຍລະອຽດລົດທີ່ຮ້ອງຂໍ, ທີ່ເຊົ່າ ແລະ ທີ່ຈອງ
Data Stores used	ຂໍ້ມູນລົດ
Description	ສະແດງລາຍລະອຽດລົດທີ່ຄົ້ນຫາ
Method	ຂໍ້ມູນລາຍລະອຽດລົດທີ່ຄົ້ນມາໄດ້ ຈະປະກອບ ດ້ວຍ ລະຫັດລົດ, ຮູບລົດ, ລຸ້ນ/ຍີ່ຫໍ້, ຄ່າເຊົ່າຕໍ່ວັນ ແລະ ຂໍ້ມູນອື່ນໆທີ່ກ່ຽວຂ້ອງ ເຊິ່ງຈະຖືກສົ່ງໄປທີ່ Process ເພື່ອຈອງ ຫຼື ເຊົ່າລົດຕໍ່ໄປ

Thank you

Q and A