**ບົດທີ 1**

# **ບົດສະເໜີ**

## ຄວາມສຳຄັນຂອງບັນຫາ.

ທຸກວັນນີ້, ເທັກໂນໂລຊີຄອມພີວເຕີ້ຖືວ່າມີບົດບາດສໍາຄັນຫຼາຍຕໍ່ວິຖີການດໍາລົງຊີວິດຂອງຄົນເຮົາ ແລະ ຕໍ່ກັບທຸກຂົງເຂດວຽກງານ ເພື່ອຍົກລະດັບຄຸນນະພາບຊີວິດການເປັນຢູ່ຂອງມວນມະນຸດ ໃຫ້ສະດວກຍີ່ງຂື້ນ ແລະ ທັງຊ່ວຍເພີ່ມປະສິດທິພາບຊີວິດການເປັນຢູ່ຂອງມວນມະນຸດໃຫ້ສະດວກຍີ່ງຂື້ນ ແລະ ທັງຊ່ວຍເພີ່ມປະສິດທີພາບໃນການເຮັດວຽກຕ່າງໆ ຂອງແຕ່ລະຂະແໜງການ ໂດຍນໍາໃຊ້ເຄື່ອງຄອມພິວເຕີ້ ທົດແທນແຮງງານຄົນ. ໃນບັນດາປະເທດທີ່ພັດທະນາແລ້ວ ເຊັ່ນວ່າ: ປະເທດຍີ່ປຸ່ນ,ສາຫະລັດອາເມລິກາ, ເກົ່າຫຼີ ແລະ ປະເທດອື່ນໆໃນໂລກ ກໍ່ໄດ້ປາຍຸກໃຊ້ໃນວຽກງານການສຶກສາ ເຮັດໃຫ້ສາມາດເພີ່ມຄຸນນະພາບການສຶກສາຂອງນັກຮຽນໃນປະເທດເຫຼົ່ານັ້ນສູງພໍສົມຄວນ ໃນນັ້ນ, ເທັກໂນໂລຊີຄອມພິວເຕີ້ ກໍໄດ້ຖືກນຳເຂົ້າມາໃຊ້ໃນວຽກງານການສືກສາຂອງປະເທດລາວເຮົາ ໄດ້ໃນລະດັບໃດໜື່ງ, ແຕ່ກໍຍັງບໍ່ທັນຕອບສະໜອງ ຕາມຄວາມຕ້ອງການຂອງວຽກງານການສຶກສາໃນ ສປປ ລາວ ໄດ້ເທົ່າທີ່ຄວນ.

ໂຮງຮຽນພອນສະຫວັນ ແລະ ຊົ່ນເຜົ່າກຽມມະຫາວິທະຍາໄລ, ມີຊື່ຫຍໍ້ວ່າ:” ຮ ພ ຊ”ເປັນໂຮງຮຽນທີ່ຂື້ນກັບສຳນັກງານອະທິການບໍດີ ພາຍໃຕ້ການຊີ້ນຳຂອງທ່ານ ຮອງອະທິການບໍດີ ມະຫາວິທະຍາໄລແຫ່ງຊາດ,ເຊີ່ງມີພາລະບົດບາດໃນການສ້າງຊັບພະຍາກອນມະນຸດ ກໍຄືການສ້າງນັກຮຽນເກັ່ງ, ໂຮງຮຽນແຫ່ງນີ້ໄດ້ຮັບການຊ່ວຍເຫຼືອລ້າຈາກລັດຖະບານ ປະເທດ ສສ ຫວຽດນາມ ລວມມູນຄ່າ 47 ຫວຽກນາມຕື້ດົ້ງ ແລະ ຖືກນຳໃຊ້ຢ່າງເປັນທາງການແຕ່ວັນທີ່ 07 ກຸມພາ 2007 ເປັນຕົ້ນມາເຊີ່ງປະກອບດ້ວຍ 07 ອາຄານຄື: ອາຄານອຳນວຍການ, ອາຄານຮຽນ, ອາຄານຫ້ອງທົດລອງ-ຫໍສະໝຸດ, ສະໂມສອນ, ໂຮງອາຫານ ແລະ ຫໍພັກ ສອງຫຼັງ, ສຳລັບຊັ້ນຮຽນແມ່ນມີແຕ່ຂັ້ນ ມ1 ເຖິງຂັ້ນ ມ7. ລະບົບການຈັດຕັ້ງປະກອບມີ ຄະນະອຳນວຍການ 04 ທ່ານ ປະກອບມີໜ່ວຍງານ, ມີຄູ-ອາຈານທັງໝົດ 56 ທ່ານ ຍິງ 29 ທ່ານ ມີນ້ອງນັກຮຽນທັງໝົດ 479 ນ້ອງ ຍິງ 282 ນ້ອງ ແລະ ໄດ້ມີຄູຈາກ ສສ ຫວຽດນາມ ມີ 01 ທ່ານ ທີ່ມາຊ່ວຍສອນ ວິຊາ ພາສາຫວຽດນາມ.ໂຮງຮຽນພອນສະຫວັນ ແລະ ຊົນເຜົ່າກຽມມະຫາວິທະຍາໄລ ເປັນອົງ​ກອນໜຶ່ງທີ່ມີ​ລັກສະນະ​ເອກະລາດ​ແຫ່ງ​ດຽວ​ໃນ​ປະ​ເທດ ທີ່​ຢູ່ໃຕ້​ການຊີ້​ນຳ ຂອງ​ມະຫາວິທະຍາ​ໄລ​ແຫ່ງ​ຊາດ, ​ເຊິ່ງມີຖານະທຽບ​ເທົ່າ​ກັບພາກວິຊາ, ​ສູນ​ການ​ສຶກສາ ຫຼື ບັນດາຫ້ອງ​ການຕ່າງໆຂອງມະຫາວິທະຍາ​ໄລ​ແຫ່ງ​ຊາດ. ລະບົບຄຸ້ມຄອງການຮຽນ-ການສອນ ຂອງໂຮງຮຽນພອນ​ສະຫວັນມະຫາວິທະຍາໄລແຫ່ງຊາດ ມີຈຸດປະສົງເພື່ອ ປະຍຸກໃຊ້ຄວາມສາມາດ ທາງດ້ານເທັກໂນໂລຊີຄອມພິວເຕີ້ມາຊ່ວຍເພີ່ມປະສິດທິພາບໃນການຈັດເກັບຂໍ້ມູນ, ໂດຍຈະຈັດເກັບຂໍ້ມູນຕ່າງໆ ໃນຮູບແບບຂໍ້ມູນດີຈີຕໍ, ຊ່ວຍການເກັບກໍາຂໍ້ມູນນັກຮຽນ ແລະ ຄູ, ຊ່ວຍບັນທຶກການຂື້ນຫ້ອງ-ລົງຫ້ອງຂອງນັກຮຽນ, ຄູອາຈານ, ພ້ອມທັງຊ່ວຍໃຫ້ຂະບວນການລົງທະບຽນງ່າຍ ແລະ ໄວຂື້ນກ່ວາເກົ່າ. ນອກນັ້ນຍັງສາມາດຈັດເກັບຄະແນນຂອງນັກຮຽນ ໄດ້ອີກດ້ວຍ. ໃນເວລາດຽວກັນ, ມັນກໍ່ຍັງສາມາດຄົ້ນຫາຂໍ້ມູນຕ່າງໆ ໄດ້ໄວຍີ່ງຂື້ນ ແລະ ລາຍງານຂໍ້ມູນຕ່າງໆ ໃຫ້ແກ່ຄະນະອຳນວຍການໄດ້ສະດວກວ່ອງໄວຂື້ນ. ເນື່ອງຈາກວ່າໂຮງຮຽນພອນ​ສະຫວັນ ຍັງບໍທັນໄດ້ນໍາໃຊ້ໂປແກມທີ່ທັນສະໄໝ ເຂົ້າມາຊ່ວຍດຳເນີນວຽກງານຂອງເຂົາເຈົ້າ ເຊີ່ງການເກັບກຳຂໍ້ມູນ ແລະ ເຮັດບົດລາຍງານຕໍ່ຄະນະອຳນວຍການ ແມ່ນຍັງບັນທຶກເປັນແຟ່ມເອກະສານ ແລະ ມີບາງຄົນກໍບັນທຶກເປັນ Word/Excel ມາຊ່ວຍວຽກບັນທຶກຂໍ້ມູນບາງສ່ວນ. ດັ່ງນັ້ນຈື່ງເຮັດໃຫ້ການຈັດເກັບຂໍ້ມູນຕ່າງໆບໍເປັນລະບຽບ, ການບັນທຶກຄະແນນ ແລະ ປະເມີນຜົນການຮຽນຂອງນັກຮຽນ ຕ້ອງໃຊ້ເວລາດົນ, ການຄົ້ນຫາຂໍ້ມູນຍັງຊ້າບໍທັນໃຈ

ດັ່ງນັ້ນ ພວກຂ້າພະເຈົ້າຈື່ງເກີດມີແນວຄິດຢາກພັດທະນາ ລະບົບຄຸ້ມຄອງການຮຽນ-ການສອນຂອງໂຮງຮຽນພອນ​ສະຫວັນ ມະຫາວິທະຍາໄລແຫ່ງຊາດ ນີ້ຂື້ນເພື່ອຊ່ວຍແກ້ໄຂບັນຫາຕ່າງໆ ທີ່ໄດ້ກ່າວມາຂ້າງເທີງນັ້ນ. ອີກດ້ານໜື່ງ, ກໍ່ແມ່ນການພັດທະນາລະບົບຂື້ນມາເພື່ອແບ່ງເບົາພາລະ ແລະ ຫຼຸດຜ່ອນເວລາການເຮັດວຽກຂອງພະນັກງານທີ່ກ່ຽວຂ້ອງກັບລະບົບດັ່ງກ່າວ.

## **ຈຸດປະສົງຂອງການສຶກສາ**

* ເພຶ່ອສຶກສາບັນຫາກ່ຽວກັບການຈັດການກັບຂໍ້ມູນຕ່າງໆຂອງໂຮງຮຽນພອນ​ສະຫວັນ ມະຫາວິທະຍາ ໄລແຫ່ງຊາດ ການເກັບກຳຂໍ້ມູນພື້ນຖານ, ຂະບວນການລົງທະບຽນ, ການຕິດຕາມການຂື້ນຫ້ອງຂອງນັກຮຽນ ແລະ ຄູອາຈານ, ການຈັດເກັບຄະແນນ, ການຈັດອັນດັບທີ່ ແລະ ການປະເມີນຜົນການຮຽນຂອງນັກຮຽນ ແລະ ຄູອາຈານ, ສາມາດຄົ້ນຫາຂໍ້ມູນ ແລະ ລາຍງານຂໍ້ມູນຕ່າງໆ ໄດ້ໄວ

ກ່ວາເກົ່າ.

* ເພື່ອສ້າງ Web Application ລະບົບຄຸ້ມຄອງການຮຽນ-ການສອນຂອງໂຮງຮຽນ ພອນ​ສະຫວັນ ມະຫາວິທະຍາໄລແຫ່ງຊາດ.
* ເພື່ອແກ້ໄຂບັນຫາ ກ່ຽວກັບການຈັດການຂໍ້ມູນຕ່າງໆທີ່ບໍ່ເປັນລະບົບ, ການເກັບກໍາຂໍ້ມູນພະນັກງານຄູ, ການລົງທະບຽນຊັກຊ້າ ການລາຍງານຂໍ້ມູນທີ່ບໍ່ທັນວ່ອງໄວເທົ່າທີ່ຄວນ ແລະ ບັນຫາການເຮັດໃບຢັ້ງຢືນນັກຮຽນ ແລະ ໃບຄະແນນສະບັບພາສາອັງກິດແມ່ນກຳລັງພັດທະນາ.

## **ຂອບເຂດການຄົ້ນຄວ້າ**

ລະບົບຄຸ້ມຄອງການຮຽນ-ການສອນ ຂອງໂຮງຮຽນພອນສະຫວັນ ແລະ ຊົນເຜົ່າກຽມມະຫາວິທະຍາໄລແມ່ນລະບົບ Client Server Web Application ເຊີ່ງປະກອບດ້ວຍຫຼາຍໜ້າວຽກຫຼັກດັ່ງລຸ່ມນີ້:

* ຈັດການຂໍ້ມູນພື້ນຖານ ( ຂໍ້ມູນນັກຮຽນ, ຂໍ້ມູນຄູອາຈານ, ຂໍ້ມູນຊັ້ນຮຽນ, ຂໍ້ມູນຫ້ອງຮຽນ, ຂໍ້ມູນວິຊາຮຽນ, ຂໍ້ມູນອາຈານສອນ, ຂໍ້ມູນຜູ້ໃຊ້, ຂໍ້ມູນຄະແນນ, ຂໍ້ມູນວັນຂາດຮຽນ,ຂໍ້ມູນໃບຢັ້ງຢືນນັກຮຽນ ສະບັບພາສາລາວ,ພາສາອັງກິດ).
* ລົງທະບຽນ ( ກວດສອບ ແລະ ລົງທະບຽນ).
* ຈັດການຂໍ້ມູນຄະແນນ( ບັນທຶກຄະແນນ ແລະ ຄົ້ນຫາຄະແນນ).
* ປະເມີນຜົນ ( ຕິດຕາມນັກຮຽນ, ຈັດອັນດັບທີ່, ເລື່ອນຊັ້ນ).
* ລາຍງານ ( ຊີວະປະຫວັດນັກຮຽນ, ສະຖິຕິນັກຮຽນ, ວັນຂາດນັກຮຽນ, ໃບຢັ້ງຢືນນັກຮຽນສະບັບລາວ-ອັງກິດ).

## **ປະໂຫຍດທີ່ຄາດວ່າຈະໄດ້ຮັບ**

ໃນການສຶກສາຄົ້ນຄວ້າຄັ້ງນີ້, ລະບົບຄຸ້ມຄອງການຮຽນ-ການສອນຂອງໂຮງຮຽນພອນ​ສະຫວັນມະ ຫາວິທະຍາໄລແຫ່ງຊາດໄດ້:

* ຊ່ວຍໃຫ້ການຈັດເກັບຂໍ້ມູນເປັນລະບົບລະບຽບເທີ່ງເຄື່ອງຄອມພິວເຕີ້.
* ຊ່ວຍໃຫ້ການເກັບກຳພະນັກງານຄູເປັນລະບົບດີ ແລະ ສະດວກວ່ອງໄວຂື້ນ.
* ຊ່ວຍໃຫ້ການລົງທະບຽນຮຽນໃຫ້ນັກຮຽນໄວຂື້ນ.
* ຊ່ວຍໃຫ້ການຈັດເກັບຂໍ້ມູນຄະແນນ, ການຈັດອັນດັບທີ, ປະເມີນຜົນການຮຽນ ແລະ ປະເມີນຄູໃຊ້ເວລາ.

# **ບົດທີ 2**

# **ທົບທວນເອກະສານ ແລະ ບົດຄົ້ນຄວ້າທີ່ກ່ຽວຂ້ອງ**

* 1. **ທິດສະດີທີ່ກ່ຽວຂ້ອງ**

### ຄວາມຮູ້ພື້ນຖານກ່ຽວກັບຖານຂໍ້ມູນ

Databases ຫຼື ຖານຂໍ້ມູນຄືກຸ່ມຂອງຂໍ້ມູນທີ່ຖືກເກັບລວບລວມໄວ້ ໂດຍມີຄວາມສໍາພັນເຊິ່ງກັນ ແລະ ກັນໂດຍບໍ່ໄດ້ບັງຄັບວ່າຂໍ້ມູນທັງໝົດນີ້ ຈະຕ້ອງເກັບໄວ້ໃນແຟ້ມຂໍ້ມູນດຽວກັນ ຫຼື ແຍກເກັບຫຼາຍໆແຟ້ມຂໍ້ມູນ

ລະບົບຖານຂໍ້ມູນ ຄືລະບົບທີ່ລວບລວມຂໍ້ມູນຕ່າງໆ ທີ່ກ່ຽວຂ້ອງກັນເຂົ້າໄວ້ດ້ວຍກັນຢ່າງມີລະບົບ,ມີຄວາມສໍາພັນລະຫວ່າງຂໍ້ມູນຕ່າງໆທີ່ຊັດເຈັນໃນລະບົບຖານຂໍ້ມູນຈະປະກອບດ້ວຍແຟ້ມຂໍ້ມູນຫຼາຍແພ້ມທີ່ມີຂໍ້ມູນກ່ຽວຂ້ອງກັນ,ສໍາພັນກັນເຂົ້າໄວ້ດ້ວຍກັນຢ່າງເປັນລະບົບ ແລະ ເປີດໂອກາດໃຫ້ ຜູ້ໃຊ້ສາມາດໃຊ້ງານ ແລະ ຮັກສາປ້ອງກັນຂໍ້ມູນເຫຼົ່ານີ້ໄດ້ຢ່າງມີປະສິດທິພາບ ໂດຍມີຊອບແວຣ໌ທີ່ປຽບສະເໝືອນສື່ກາງລະຫວ່າງຜູ້ໃຊ້ ແລະ ໂປຣແກຣມຕ່າງໆ ທີ່ກ່ຽວຂ້ອງກັບການໃຊ້ຖານຂໍ້ມູນ ເອີ້ນວ່າລະບົບຈັດການຖານຂໍ້ມູນ ຫຼື DBMS (Databases Management System)ມີໜ້າທີ່ຊ່ວຍໃຫ້ຜູ້ໃຊ້ເຂົ້າເຖິງຂໍ້ມູນໄດ້ງ່າຍສະດວກ ແລະ ມີປະສິດທິພາບການເຂົ້າເຖິງຂໍ້ມູນຂອງຜູ້ໃຊ້ອາດເປັນການສ້າງຖານ ຂໍ້ມູນ,ການແກ້ໄຂຖານຂໍ້ມູນ ຫຼື ການຕັ້ງຄໍາຖາມເພື່ອໃຫ້ໄດ້ຂໍ້ມູນມາໂດຍຜູ້ໃຊ້ບໍ່ຈໍາເປັນຕ້ອງຮັບຮູ້ກ່ຽວກັບລາຍລະອາດພາຍໃນໂຄງສ້າງຂອງຖານຂໍ້ມູນ.

### ຄຸນລັກສະນະຂອງຖານຂໍ້ມູນ

* ລົດຄວາມຊ້ຳຊ້ອນຂອງຖານຂໍ້ມູນໃຫ້ເຫຼືອນ້ອຍທີ່ສຸດ (Minimum Redundancy)
* ມີຄວາມຖືກຕ້ອງສູງສຸດ (Maximum Integrity)
* ມີຄວາມເປັນອິດສະລະຂອງຂໍ້ມູນ (Data Independence)
* ມີລະບົບຄວາມປອດໄພຂອງຂໍ້ມູນສູງ (High Degree of Data Security)
* ການຄວບຄຸມຖານຂໍ້ມູນຈະຢູ່ສ່ວນກາງ (Logically Centralized Control)
  + - 1. **ຄວາມສໍາຄັນຂອງລະບົບຖານຂໍ້ມູນ**
* ສາມາດລົດຄວາມຊ້ຳຊ້ອນຂອງຂໍ້ມູນໄດ້
* ຫຼີກລ້ຽງຄວາມຂັດແຍ້ງຂອງຂໍ້ມູນໄດ້
* ສາມາດໃຊ້ຂໍ້ມູນຮ່ວມກັນໄດ້
* ສາມາດກໍານົດຄວາມເປັນມາດຕະຖານດຽວກັນຂອງຂໍ້ມູນໄດ້
* ສາມາດກໍານົດລະບົບຄວາມປອດໄພຂອງຂໍ້ມູນໄດ້
* ສາມາດເກີດຄວາມເປັນອິດສະຫຼະຂອງຂໍ້ມູນໄດ້

### ຫນ້າທີ່ຂອງລະບົບຖານຂໍ້ມູນ

* ຫນ້າທີ່ຂອງລະບົບຖານຂໍ້ມູນມີຄື:
* ຈັດການພົດຈະນານຸກົມຂໍ້ມູນ
* ການຈັດເກັບຂໍ້ມູນ
* ການແປງ ແລະ ນໍາສະເໜີຂໍ້ມູນ
* ການຈັດການລະບົບຄວາມຫມັ້ນຄົງ
* ການຄວບຄຸມການເຂົ້າເຖິງຂໍ້ມູນຂອງຜູ້ໃຊ້ຫຼາຍຄົນ
* ເກັບສໍາຮອງ ແລະ ກູ້ຄືນຂອງຂໍ້ມູນ
* ຄວບຄຸມຄວາມຖືກຕ້ອງຂອງຂໍ້ມູນ
* ພາສາທີ່ໃຊ້ໃນການເຂົ້າເຖິງຂໍ້ມູນ ແລະ ການເຊື່ອມຕໍ່ກັບໂປຣແກຣມປະຍຸກ
* ການຕິດຕໍ່ສື່ສານກັບຖານຂໍ້ມູນ

### ສະຖາປັດຕະຍະກໍາລະບົບຖານຂໍ້ມູນ

ສະຖາປັດຕະຍະກໍາລະບົບຖານຂໍ້ມູນ ມີ 3 ລະດັບຄື:

* ລະດັບພາຍໃນ (Internal Level)
* ລະດັບລັກສະນະແນວຄິດ (Conceptual Level)
* ລະດັບພາຍນອກ (External Level)

1. ລະດັບພາຍໃນ (Internal Level)

ເປັນການນໍາເອົາຂໍ້ມູນທີ່ໄດ້ຈາກລະດັບແນວຄິດມາວິເຄາະ ແລະ ອອກແບບໂດຍແປງໃຫ້ຢູ່ໃນຮູບແບບຂອງການຈັດການຖານຂໍ້ມູນ (DBMS) ທີ່ເລືອກໃຊ້ໂດຍເສີມແນວຄິດການເຮັດ Normalization car Demoralization.

1. ລະດັບລັກສະນະແນວຄິດ (Conceptual Level)

ເປັນການນໍາເອົາຂໍ້ມູນທີ່ໄດ້ຈາກການວິເຄາະຄວາມຕ້ອງການຂອງຜູ້ໃຊ້ຂໍ້ມູນໃນລະດັບພາຍນອກມາອອກແບບຖານຂໍ້ມູນເພື່ອໄດ້ໂຄງຮ່າງຂອງຖານຂໍ້ມູນໃນລະດັບແນວຄິດທີ່ປະກອບດ້ວຍໂຄງສ້າງຂອງຖານຂໍ້ມູນສ່ວນຫນຶ່ງເປັນຄວາມສໍາພັນກັນ.

1. ລະດັບພາຍນອກ (External Level)

ເປັນການນໍາເອົາຂໍ້ມູນທີ່ໄດ້ຈາກລະດັບພາຍໃນມາກໍານົດໂຄງສ້າງຂໍ້ມູນ ແລະ ການຈັດເກັບວິທີການເຂົ້າເຖິງການຈັດການດ້ານລະບົບຄວາມປອດໄພຂອງຂໍ້ມູນເພື່ອຖານຂໍ້ມູນເຮັດວຽກໄດ້ຢ່າງມີປະສິດທິພາບ.

* + 1. **ຄວາມຮູ້ພື້ນຖານກ່ຽວກັບພາສາ NOSQL**

NoSQL ຫຼື “non SQL”, “non relational” ,”not only SQL” ໝາຍ​ເຖິງ​ການທີ່ [database](https://saixiii.com/what-is-database/)ໃຊ້​ວິທີ​ການສ້າງ​ຖານ​ຂໍ້​ມູນ​ ແລະ ​ການ​ດຶງ​ຂໍ້​ມູນທີ່​ແຕ່ກຕ່າງຈາກ​ຮູບ​ແບບ​ປົກກະຕິ ([SQL](https://saixiii.com/sql-statement/)) ທີ່​ອອກ​ມາ​ ໃນ​ຮູບ​ແບບ​ຕາ​ຕະລາງ​ເຊັ່ນ​ດຽວ​ກັບ [relational databases](https://saixiii.com/what-is-database/)  ຊຶ່ງ​ຕັ້ງ​ແຕ່​ຊ່ວງ 1960 ທີ່ database ໄດ້​​ກຳ​ເນີດ​ຂຶ້ນ​ແຕ່ NoSQL ກໍ​ຢັງ​ບໍ່​ເປັນ​ທີ່​ນິຍົມ ຈົນ​ກະ​ທັງ​ຊ່ວງ​ປີ 2000 NoSQL ເລີ່ມ​ຖືກ​ເພີ່ມ​ເຂົ້າ​ມາ​ໃນ Web2.0 ເຮັດໃຫ້​ຫຼາຍ​ບໍລິສັດ​ຊັ້ນ​ນຳ ​ເຊັ່ນ Facebook, Google ແລະ Amazon.com ໄດ້​ເລີ່ມ​ນຳ​ມາ​ໃຊ້ ສ່ວນ​ຫຼາຍ​ຖືກ​ນຳ​ມາ​ໃຊ້​ງານ​ກ່ຽວກັບ big data ແລະ real-time web applications ນັ້ນ​ຄື​ທີ່​ມາ​ຂອງ NoSQL ໂດຍ​ປົກກະຕິ​ແລ້ວ NoSQL ງ່າຍ​ຕໍ່​ການ​ຂະຫຍາຍ​ລະບົບທີ່​ເປັນ​ຮູບ​ແບບ cluster (ແນ່​ນອນ) ຊຶ່ງ​ນີ້​ເປັນ​ຈຸດ​ອ່ອນ​ຂອງ relational database ຫຼັກ​ການ​ຂອງ NoSQL ຄື​ໃຊ້​ໂຄງ​ສ້າງ​ຂອງ​ຂໍ້​ມູນ​ໄດ້​ຫຼາຍ​ຮູບ​ແບບ​ key-value, wide column, graph ຫລື document ຊຶ່ງ​ຈະ​ແຕກ​ຕ່າງ​ຈາກ​ຮູບ​ແບບ​ຂອງ relational database ຊຶ່ງ​ບາງເທື່ອ​ເຮັດໃຫ້ NoSQL ມີ​ຄວາມ​ວ່ອງໄວ​ຫລາຍກວ່າ.

* **ປະ​ເພດ​ຂອງ NoSQL**

ມີ​ການ​ຈຳ​ແນກ​ໄດ້​ຫຼາຍ​ຮູບ​ແບບ​ສຳລັບ NoSQL database ໂດຍ​ແຕ່​ລະ​ຮູບ​ແບບ​ໄດ້​ມີ​ການ​ຈັດ​ກລຸ່ມ ​ແລະ​ກຸ່ມ​ຍ່ອຍ​ໄວ້ ບາງເທື່ອ​ອາດຈະ​ມີ​ບາງ​ຊະນິດ​ກາບກ່ຽວ​ກັນ​ດ້ວຍເຊັ່ນ:

* **Column**: Accumulo, Cassandra, Druid, HBase, Vertica, SAP HANA
* **Document**: Apache CouchDB, ArangoDB, Clusterpoint, Couchbase, DocumentDB, HyperDex, IBM Domino, MarkLogic, MongoDB, OrientDB, Qizx, RethinkDB
* **Key-value**: Aerospike, ArangoDB, Couchbase, Dynamo, FairCom c-treeACE, FoundationDB, HyperDex, InfinityDB, MemcacheDB, MUMPS, Oracle NoSQL Database, OrientDB, Redis, Riak, Berkeley DB
* **Graph**: AllegroGraph, ArangoDB, InfiniteGraph, Apache Giraph, MarkLogic, Neo4J, OrientDB, Virtuoso, Stardog
* **Multi-model**: Alchemy Database, ArangoDB, CortexDB, Couchbase, FoundationDB, InfinityDB, MarkLogic, OrientDB

**ກ. Key-value store**

ເຮັດວຽກ​ໃນ​ຮູບ​ແບບ array (Map ຫລື Dict) ຊຶ່ງ​ເປັນ​ໂຄງ​ສ້າງ​ພື້ນ​ຖານ​ທີ່​ສ້າງ​ຄວາມ​ສຳພັນ​ເປັນ​ຄູ່ ຊຶ່ງ Key ຈະ​ຕ້ອງ​ບໍ່​ຊ້ຳ​ກັນ ຮູບ​ແບບ​ນີ້​ຄື​ແບບ​ທີ່​ງ່າຍ​ທີ່ສຸດ ແລະ ມີ​ຄວາມ​ສາມາດ​ໃນ​ການ​ປະ​ມວມ​ຜົນ​ທີ່​ສູງ​ຫຼາຍ.

**ຂ. Document store**

ຮູບ​ແບບ​ຂອງ document store ຄື​ການ ເຂົ້າ​ລະ​ຫັດ​ຂໍ້​ມູນ​ດ້ວຍ​ຮູບ​ແບບ format ມາດຕາ​ຖານ​ເຊັ່ນ: [XML](https://saixiii.com/what-is-xml/),YAML,[JSON](https://saixiii.com/what-is-json/) ຫຼື binary ແລະ​ ເກັບ​ໄວ້​ໃນ database ດ້ວຍ unique key ທີ່​ເຊື່ອມ​ກັບ​ຂໍ້​ມູນ ຄ້າຍ​ຮູບ​ແບບ​ຂອງ key-value ໂດຍ database ຈະ​ມີ API ຫລື ພາສາ​ທີ່​ໃຊ້ query ຂໍ້​ມູນສາມາດ​ຕິດ​ຕັ້ງ​ໄດ້​ຫຼາຍ​ຮູບ​ແບບ ຕາມ​ກຸ່ມ documents

* Collections
* Tags
* Non-visible metadata
* Directory hierarchies

ຖ້າ​ທຽບ​ກັບ relational database ຈະ​ເຫັນ​ວ່າແຕ່​ລະ record ສາມາດ​ມີ​ຈຳນວນ field ບໍ່​ເທົ່າ​ກັນ​ໄດ້ຊຶ່ງ​ບໍ່​ຄືກັບ​ຮູບ​ແບບ​ຕາ​ຕະລາງ.

**ຄ. Graph**

ຖືກ​ອອກ​ແບບ​ສຳລັບ​ຂໍ້​ມູນ​ທີ່​ມີ​ຕ້ອງ​ສະແດງ​ເປັນ graph ທີ່​ປະກອບ​ໄປ​ດ້ວຍ​ຂໍ້​ມູນ​ຕັວ​ເລກ​ນັບ​ບໍ່​ຖ້ວນ ເຊັ່ນ: social relations, link ຂອງ​ການ​ຂົ່ນ​ສົ່ງ, ຖະໜົນ​ແຜນ​ທີ່ ຫຼື ລະບົບ network

* + 1. **ຄວາມຮູ້ກ່ຽວກັບພາສາ HTML**

HTML ແມ່ນພາສາທີ່ໃຊ້ໃນການຂຽນ Web Page ທີ່ສາມາດສະແດງຜົນເທິງ Website ຫຼື ທີ່ເຮົາເອີ້ນວ່າ: Web page ຖືກພັດທະນາ ແລະ ກຳນົດມາດຕະຖານທີ່ແນ່ນອນໂດຍອົງກອນ world wide web Consortium ( W3C) ຖືກພັດທະນາທາງດ້ານ Software ຂອງ Microsoft ເຮັດໃຫ້ HTML ເປັນອີກພາສາໜື່ງທີ່ໃຊ້ໃນການຂຽນ Program Dreamweaver, Notepad, Edit plus, ສ່ວນການເອີ້ນໃຊ້ງານ ຫຼື ທົດສອບການເຮັດວຽກຂອງເວບຈະໃຊ້ໂປຣແກຣມ web Browser ເຊັ່ນ: IE Microsoft, Internet Explorer(IE), Google Chrome, Mozilla Firefox, Safari, opera, Netscape Navigator ເປັນຕົ້ນ.

ຈຸດເດັ່ນຂອງ HTML ປະກອບມີ: Semantic Markup ( ໂຄດເປັນລະບຽບເຮັດໃຫ້ search Engine ເກັຍຂໍ້ມູນໄດ້ງ່າຍ), from Enhancement ( ເພີ່ມປະສິດທິພາບຂອງຟອມ), ສຽງ ແລະ ວີດີໂອ.

* + 1. **ຄວາມຮູ້ກ່ຽວກັບພາສາ CSS**

CSS ( Cascading Style Sheet) ເປັນພາສາທີ່ໃຊ້ຈັດການຮູບແບບເອກະສານ HTML ໃຫ້ມີຄວາມສວຍງາມ, ມີຄຸນສົມບັດເຊັ່ນ: ການກຳນົດຂອບ, ສີ, ເສັ້ນ, ພື້ນຫຼັງ, ໄລຍະຫ່າງ ແລະ ຮູບແບບຕົວໜັງສືຕ່າງໆເຊັ່ນ: ການສະແດງຜົນເທີງໜ້າຈໍພາບ, ການສະແດງຜົນເທີງມືຖື ແລະ ຊ່ວຍຫຼຸດການໃຊ້ພາສາ HTML ໃນການຕົກແຕ່ງ Webpage ເຮັດໃຫ້ Code ເຫຼືອພຽງສ່ວນເນື້ອໃນທີ່ເຂົ້າໃຈງ່າຍ, ການແກ້ໄຂງ່າຍຂື້ນ ແລະ ຂະໜາດຟ່າຍນ້ອຍລົງ,ດາວໂຫຼດໄດ້ງ່າຍຂື້ນ ແລະ ທີ່ສຳຄັນຍັງສາມາດຄວບຄຸມການສະແດງຜົນໃຫ້ຄືກັນໄດ້ຫຼາຍໆ Web Browser.

* + 1. **ຄວາມຮູ້ກ່ຽວກັບພາສາ JavaScript**

ພາສາ JavaScript ຫລື​ ຫຍໍ້ JS ເປັນ​ພາສາ​ຂຽນ​ໂປຣ​ແກຣມ​ທີ່​ຖືກ​ພັດທະນາ ​ແລະ​ ປະຕິບັດ​ຕາມ​ຂໍ້​ກຳນົດ​ມາດຕະຖານ​ຂອງ ECMAScript; ພາສາ JavaScript ນັ້ນ​ເປັນ​ພາສາ​ລະ​ດັບ​ສູງ ຄອມ​ພາຍ​ໃນ​ຂະນະ​ທີ່​ໂປຣ​ແກຣມ​ຣັນ (JIT) ແລະ​ ເປັນ​ພາສາ​ຂຽນ​ໂປຣ​ແກຣມ​ແບບ​ຫຼາຍ​ຂະ​ບວນ​ເຊັ່ນ: ການ​ຂຽນ​ໂປຣ​ແກຣມ​ແບບ​ຂັ້ນ​ຕອນ, ການ​ຂຽນ​ໂປຣ​ແກຣມ​ແບບ​ວັດ​ຖຸ, ພາສາ JavaScript ມີ​ໄວ​ຍະກອນທີ່​ຄື​ກັບ​ພາສາ C ໃຊ້​ວົງ​ເລັບ​ເພື່ອ​ກຳນົດ​ບອກ​ຂອງ​ຄຳ​ສັ່ງ ນອກ​ຈາກ​ນີ້ JavaScript ຢັງ​ເປັນ​ພາສາ​ທີ່​ມີ​ປະ​ເພດ​ຂໍ້​ມູນ​ແບບ​ Dynamic​ ເປັນ​ພາສາ​ແບບ Prototype-based ແລະ First-class function.

ພາສາ JavaScript ນັ້ນ​ຖື​ວ່າ​ເປັນ​ເທັກ​ໂນ​ໂລ​ຊີ ​ຫຼັກ​ຂອງ​ການ​ພັດທະນາ​ເວັບ​ໄຊ​ (World Wide Web) ມັນ​ເຮັດໃຫ້​ໜ້າ​ເວັບ​ ສາມາດ​ຕອບ​ໂຕ້​ກັບ​ຜູ້​ໃຊ້​ໄດ້ ​ໂດຍ​ທີ່​ບໍ່​ຈຳ​ເປັນ​ຕ້ອງ​ລີເຟດໜ້າ​ໃໝ່ (Dynamic website) ເວັບ​ໄຊ​ຈຳນວນ​ຫຼາຍ​ໃຊ້​ພາສາ JavaScript ສຳລັບ​ຄວບ​ຄຸມ​ການ​ເຮັດ​ວຽກງານ​ທີ່​ດ້ານ Client-side ນັ້ນ​ເຮັດໃຫ້​ເວັບ​ເບ​ຣາ​ວ​ເຊີຕ່າງ​ໆ ມີ JavaScript engine ທີ່​ໃຊ້​ສຳລັບ​ປະ​ມວນ​ຜົນສະຄິບ​ຂອງ​ພາສາ JavaScript ທີ່​ຣັນ​ເທິງ​ເວັບ​ບາວ​ເຊີເນື່ອງ​ຈາກ​ພາສາ JavaScript ເປັນ​ພາສາ​ຂຽນ​ໂປຣ​ແກຣມ​ແບບ​ຫຼາຍ​ຮູບແບບ ເຮັດໃຫ້​ມັນ​ຮອງ​ຮັບ​ການ​ຂຽນ​ໂປຣ​ແກຣມ​ທັງ​ແບບ Event-driven, Functional ແລະ ​ແບບ​ລຳ​ດັບ​ຂັ້ນ​ຕອນ ມັນ​ມີ​ Library (APIs) ສຳລັບ​ເຮັດວຽກ​ກັບ​ຂໍ້ຄວາມ, ວັນທີ, Regular expression ແລະ ​ໂຄງ​ສ້າງ​ຂໍ້​ມູນ​ພື້ນ​ຖານ​ຢ່າງ Array ແລະ Map ຫຼື ​ແມ່ນ​ທັງ Document Object Model (DOM) ຊຶ່ງ​ເປັນ API ທີ່​ໂດຍ​ທົ່ວ​ໄປ​ແລ້ວ​ສາມາດ​ໄດ້​ເທິງ​ເວັບບາວເຊີ.

ຢ່າງໃດ​ກໍ​ຕາມ ຕົວ​ຂອງ​ພາສາ JavaScript ເອງ​ບໍ່​ໄດ້​ມີ​ຟັງ​ຊັນ​ສຳລັບອິນພຸດ/ເອົາພຸດ(I/O) ທີ່​ມາ​ກັບ​ພາສາເຊັ່ນ ຟັງ​​ຊັນ​ກ່ຽວກັບ Network ວຽກກ່ຽວ​ກັບ​ໄຟ​ລ ຫຼື Library​ ກ່ຽວກັບ​ກຣາ​ຟິກ ໂດຍ​ທົ່ວ​ໄປ​ແລ້ວ​ສິ່ງເຫຼົ່ານີ້​ຈະ​ຖືກ​ໃຫ້​ມາ​ໂດຍ Host environment (ສະພາບ​ແວດ​ລ້ອມ​ທີ່​ໃຊ້​ຣັນ​ພາສາ JavaScript) ເຊັ່ນ ເວັບ​​ເວັບບາວເຊີ ຫຼື Node.js ຊຶ່ງ​ຈະ​ແຕກ​ຕ່າງ​ກັນ​ອອກ​ໄປ ​ຕົວ​ຢ່າງ​ເຊັ່ນ ການ​ຮັບ​ຄ່າ​ໃນ​ເວັບ​​ເວັບບາວເຊີ​ຈະ​ຜ່ານ​ຟັງ​ຊັນ prompt ຊຶ່ງ​ເປັນ​ສ່ວນຫນຶ່ງ​ຂອງ Browser Object Model (BOM) ຫຼື ​ຮັບ​ຄ່າ​ຈາກ HTML ຟອມຊຶ່ງ​ເປັນ​ສ່ວນຫນຶ່ງ​ຂອງ Document Object Model (DOM) ໃນຂະນະ​ທີ່​ເທິງ Node.js ເຮົາ​ສາມາດ​ຮັບ​ຄ່າ​ໄດ້​ຈາກ Input/Output Stream ຂອງ Command line ເຖິງ​ແມ່ນ​ວ່າ​ມັນ​ຈະ​ມີ​ຄວາມ​ຄ້າຍ​ຄືກັນ​ລະຫວ່າງ​ພາສາ Java ແລະ JavaScript ເຊັ່ນ: ຊື່​ຂອງ​ພາສາ ໄວ​ຍະກອນ ຫຼືLibrary​ ​ມາດຕະຖານ​ຕ່າງ​ໆ ຢ່າງໃດ​ກໍ​ຕາມທັງ​ສອງ​ພາສາ​ແຕກ​ຕ່າງ​ກັນ​ຢ່າງ​ສິ້ນ​ເຊີງ​ ໃນ​ແງ່​ຂອງ​ການ​ອອກ​ແບບພາສາ Java ເປັນ​ພາສາ​ທີ່​ມີ​ປະ​ເພດ​ຂໍ້​ມູນ​ແບບ​ຄົງ​ທີ່ (Static-typing) ໃນ​ຂະນະ​ທີ່​ພາສາ JavaScript ມີ​ປະ​ເພດ​ຂໍ້​ມູນ​ແບບ​ໄດ​ນາ​ມິກ​ (Dynamic-typing) ພາສາ Java ຖືກ​ຄອມ​ພາຍເປັນ Byte-code ກ່ອນທີ່ຈະ​ລັນ ໃນ​ຂະນະ​ທີ່​ພາສາ JavaScript ຈະ​ຄອມ​ພາຍໃນ​ຕອນ​ທີ່​ໂປຣ​ແກຣມ​ລັນພາສາ Java ເປັນ​ພາສາ​ແບບ Class-based ໃນ​ຂະນະ​ທີ່​ພາສາ JavaScript ເປັນ​ພາສາ​ແບບ Prototypebased



**ຮູບທີ 10**: ພາສາ JavaScript

**ກ. ປະຫວັດ​ຄວາມ​ເປັນ​ມາ​ຂອງ​ພາສາ JavaScript**

ພາສາ JavaScript ຖືກ​ອອກ​ແບບ ​ແລະ ​ສ້າງ​ໂດຍ Brendan Eich ສຳລັບ​ເປັນ​ພາສາ​ສະຄິບ​ທີ່​ເຮັດວຽກ​ເທິງ​ເວັບ​ເບ​ຣາ​ວ​ເຊີ Navigator ທີ່​ເປັນ​ຜະລິດຕະພັນ​ຂອງ​ບໍລິສັດ Netscape ເພື່ອ​ເຮັດໃຫ້​ໜ້າ​ເວັບ​ທີ່​ໃນ​ຕອນ​ທຳອິດ​ນັ້ນ​ເປັນ​ແບບ Static ສາມາດ​ຕອບ​ໂຕ້​ກັບ​ຜູ້​ໃຊ້​ໄດ້​ໂດຍ​ທີ່​ບໍ່​ຈຳ​ເປັນ​ຕ້ອງ​ຣີ​ເຟ​ດ​ໜ້າ

​ໃໝ່ (Dynamic) ເຊັ່ນ ການ​ສົ່ງ​ຂໍ້​ມູນ​ເບື້ອງ​ຫຼັງ​ໄປ​ຢັງ Server ແລະ​ ລໍຖ້າ​ຮັບ​ຜົນ​ຕອບ​ກັບ​ມາ​ດ້ວຍ AJAX; ພາສາ JavaScript ໄດ້​ຖືກ​ເປີດ​ຕົວ​ ແລະ ​ເປັນ​ສ່ວນຫນຶ່ງ​ຂອງ​ເວັບ​ບ​າ​ວ​ເຊີ Navigator ໃນ​ເດືອນ​ກັນຍາ 1995 ໂດຍ​ໃຊ້​ຊື່​ວ່າ LiveScript ແລະ ​ໄດ້​ປ່ຽນ​ເປັນ JavaScript ໃນ​ອີກ​ສາມ​ເດືອນ​ຕໍ່​ມາ.

ໃນ​ເດືອນ​ພ​ະຈິກ 1996 Netscape ໄດ້​ສົ່ງ​ພາສາ JavaScript ໄປ​ຢັງ ECMA International ເພື່ອ​ເປັນ​ຈຸດ​ເລີ່ມ​ຕົ້ນ​ສຳລັບ​ກຳນົດ​ມາດຕະຖານ​ໃຫ້​ທຸກ​ເວັບ​ບາວເຊີປະຕິບັດຕາມ​ມາດຕະຖານ​ ດັ່ງ​ກ່າວ ເພື່ອ​ໃຫ້​ການ​ພັດທະນາ JavaScript engine ເປັນ​ໄປ​ໃນ​ທິດ​ທາງ​ດຽວ​ກັນ ນັ້ນ​ໃຫ້​ເກີດ​ການ​ເປີດ​ຕົວ​ຢ່າງ​ເປັນ​ທາງ​ການ​ສຳລັບ​ຂໍ້​ກຳນົດ​ມາດຕະຖານ ECMAScript ໃນ​ເດືອນ​ມິ​ຖຸ​ນາ 1997 ໃນ​ຊ່ວງ​ເວລາ​ຫຼັງ​ຈາກ​ນີ້ ບໍລິສັດ​ຕ່າງ​ໆ ທີ່​ພັດທະນາ​ເວັບ​ບາ​ວ​ເຊີຕ່າງ​ກໍ​ຍັງ​ພັດທະນາ JavaScript engine ບໍ່​ເປັນ​ໄປ​ໃນ​ທິດ​ທາງ​ດຽວ​ກັນ​ເທົ່າໃດ ນັ້ນ​ເຮັດໃຫ້​ນັກ​ພັດທະນາ​ເວັບ​ຕ້ອງ​ຂຽນ​ໂຄດຫຼາຍ​ເວີຊັນ​ເພື່ອ​ໃຫ້​ເຮັດວຽກ​ໄດ້​ໃນ​ທຸກ​ເວັບ​ບ​າ​ວ​ເຊີຈົນກະທັ້ງ​ໃນ​ເ​ດືອນກໍລະກົດ 2008 ໄດ້​ມີ​ການ​ຈັດການ​ປະ​ຊຸມ​ຂຶ້ນ​ທີ່ Oslo ຈາກ​ອົງ​ກ​ອນ​ແລະ ​ຝ່າຍ​ຕ່າງ​ໆທີ່​ພັດທະນາ JavaScript engine ເຮັດໃຫ້​ເກີດ​ຂໍ້​ຕົກລົງ​ຂຶ້ນ​ໃນ​ຕົ້ນ​ປີ 2009 ເພື່ອ​ລວບລວມ​ງານ​ທີ່​ກ່ຽວ​ຂ້ອງ​ທັງ​ໝົດ​ຂອງ​ພາສາ JavaScript ແລະ ​ຊຸກຍູ້​ພາສາ​ໃຫ້​ຍ່າງ​ໄປ​ຂ້າງ​ໜ້າ ນັ້ນ​ເຮັດໃຫ້​ເກີດ​ຂໍ້​ກຳນົດ​ມາດຕະຖານ ECMAScript ເວີຊັນ​ທີ່ 5 (ES5) ອອກ​ມາ​ໃນ​ເດືອນ​ທັນວາ 2009 ແລະ ​ກ່ອນ​ໜ້າ​ນີ້​ໃນ​ປີ 2008 Google ໄດ້​ເປີດ​ຕົວ​ເວັບ​ບ​າ​ວ​ເຊີ Chrome ທີ່​ມາ​ພ້ອມ​ກັບ V8 JavaScript engine ທີ່​ມີ​ແນວ​ຄິດ​ໃນ​ການ​ພັດທະນາ​ແບບ​ຄອມ​ພາຍ​ໃນ​ຕອນ​ທີ່​ໂປຣ​ແກຣມ​ລັນ (Just-in-time compilation: JIT) ຊຶ່ງ​ມັນ​ເຮັດວຽກ​ໄດ້​ໄວ​ກວ່າ​ຫຼາຍ ເຮັດໃຫ້​ຜູ້​ພັດທະນາ​ເວບບ​າ​ວ​ເຊີ​ອື່ນ​ໆ ຕ້ອງ​ປັບ​ປຸງ JavaScript engine ຂອງ​ພວກ​ເຂົາ​ໃຫ້​ເຮັດວຽກ​ໃນ​ຮູບ​ແບບ JIT ຫຼັງ​ຈາກ​ທີ່​ພັດທະນາ​ຕໍ່​ເນື່ອງ​ມາ​ອີກ​ຫຼາຍ​ປີ ໃນ​ປີ 2015 ໄດ້​ມີ​ການ​ເພີ່ມ​ຄຸນສົມບັດ​ໃໝ່​ໆ ທີ່​ຫຼາກ​ຫຼາຍ​ເຂົ້າ​ມາ ຊຶ່ງ​ຖື​ວ່າ​ເປັນ​ການ​ປ່ຽນ​ແປງ​ເທື່ອ​ສຳຄັນ ແລະ​ເຮັດໃຫ້​ເກີດ​ຂໍ້​ກຳນົດ​ມາດຕະຖານ ECMAScript 2015 ຫຼື ​ເວີຊັນທີ່ 6 (ES6) ຈົນກະທັ້ງ​ໃນ​ປີ 2015 ຕອນ​ນີ້​ເບິ່ງ​ຄື​ວ່າ​ພາສາ JavaScript ຈະ​ພັດທະນາ​ມາ​ຈົນ​ເຖິງ​ທີ່ສຸດ​ແລ້ວ ເຮັດໃຫ້​ລະຫວ່າງ​ປີ 2016 - 2019 ເວີຊັນ​ໃໝ່​ຂອງ ECMAScript ທີ່​ຖືກ​ເຜີຍແຜ່​ອອກ​ມາ​ໃນ​ແຕ່​ລະ​ປີ​ມີ​ການ​ປ່ຽນ​ແປງ​ ແລະ​ ເພີ່ມ​ຄຸນສົມບັດ​ພຽງ​ນ້ອຍໆ​ເທົ່າ​ນັ້ນ.

**ຂ. ຄຸນສົມບັດ​ຂອງ​ພາສາ JavaScript**

ECMAScript 2015 (ES6) ເປັນ​ພາສາ JavaScript ທີ່​ຖື​ວ່າ​ພັດທະນາ​ມາ​ຈົນ​ເຖິງ​ຈຸດ​ສູງ​ສຸດ​ແລ້ວ​ກໍ​ວ່າ​ໄດ້ ມັນ​ຖືກ​ເຜີຍແຜ່​ໃນ​ເດືອນ​ມິ​ຖຸ​ນາ 2015 ຊຶ່ງ​ໃນ​ເວີຊັນ​ນີ້ ໄດ້​ເພີ່ມ​ໄວ​ຍະກອນໃໝ່​ຂອງ​ພາສາ​ຫຼວງ​ຫລາຍເຊັ່ນ ການສ້າງ​ຄາດ​ດ້ວຍ​ຄຳ​ສັ່ງ class ການສ້າງ​ໂມ​ດູນ​ ແລະ​ ໃຊ້​ງານ​ມັນ​ດ້ວຍ​ຄຳ​ສັ່ງ import ແລະ export ແລະ​ ຄຳ​ສັ່ງ​ສຳລັບ​ປະ​ກາດ​ຕົວ​ປ່ຽນ let ແລະ ​ປະ​ກາດ​ຄ່າ​ຄົງທີ const ຊຶ່ງ​ເຮັດໃຫ້​ຕົວ​ປ່ຽນສາມາດ​ມີ​ຂອບ​ເຂດ​ໃນ​ບ​ລັອກ​ທີ່​ມັນ​ຖືກ​ສ້າງ​ຂຶ້ນ​ໄດ້ ແລະ ​ສິ່ງ​ອື່ນ​ໆ ທີ່​ຖືກ​ເພີ່ມ​ເຂົ້າ​ມາ​ເປັນ​ຈຳນວນ​ຫລາຍ ເຊັ່ນ: Map, Set, Weak Map, Promise, Reflection, Proxies, Template string ແລະ ​ອື່ນ​ໆ.

ໃນ​ເດືອນ​ມິ​ຖຸ​ນາ 2016 ໄດ້​ມີ​ການ​ເປີດ​ຕັວ​ເວີຊັນ 7 ຫຼື ECMAScript 2016 (ES7) ໄດ້​ມີ​ການ​ເພີ່ມ​ຕົວ​ດຳ​ເນີນ​ການ​ຍົກ​ກຳ​ລັງ \*\* (ທີ່​ກ່ອນ​ໜ້າ​ນີ້​ເຮົາ​ຈະ​ໃຊ້​ຜ່ານ​ຟັງຊັນ Math. pow) ຄຳ​ສັ່ງawait Async ສຳລັບ​ການ​ຂຽນ​ໂປຣ​ແກຣມ​ທີ່​ເຮັດວຽກ​ບໍ່​ພ້ອມ​ກັນ ແລະ ​ຟັງຊັນ includes ຂອງ​ອາ​ເລແລະ ​ໃນ​ປະຈຸບັນ ພາສາ JavaScript ຖືກ​ພັດທະນາ​ມາ​ຈົນ​ເຖິງ ECMAScript 2020 (ES11) ຊຶ່ງ​ມີ​ການ​ປ່ຽນ​ແປງ​ທີ່​ເພີ່ມ​ຂຶ້ນ​ບໍ່​ເທົ່າໃດ​ຫຼັງ​ຈາກ ES7.

* + 1. **ຄວາມຮູ້ກ່ຽວກັບ Node Js**

Node.js ເປັນ open-source ແລະ cross-platform JavaScript runtime environment ທີ່​ກຳ​ລັງ​ໄດ້ຮັບ​ຄວາມ​ນິຍົມ​ສູງ ໂດຍ​ທົ່ວ​ໄປ​ເຮົາ​ຈະ​ໃຊ້ JavaScript ໃນ​ຝັ່ງ client ແຕ່ Node.js ເຮັດໃຫ້​ເຮົາ​ໃຊ້ JavaScript ໃນ​ຝັ່ງ Server ໄດ້​ດ້ວຍ ຊຶ່ງ Node.js ສາມມາດ run​ ໄດ້​ເທິງ platform ທີ່​ຫຼາກ​ຫຼາຍ​ທັງ Windows, Linux, Unix, Mac OS X ແລະ ອື່ນໆ.

**ຮູບທີ** 12 Node Js

ຈຸດ​ເດັ່ນ​ທີ່ສຸດ​ຂອງ Node.js ຄື​ມັນ​ເຮັດວຽກ​ແບບ asynchronous ຜູ້​ອ່ານ​ອາດຈະ​ສົງ​ໃສ​ວ່າ​ແລ້ວ​ມັນ​ດີ​ແນວໃດ? ລອງ​ມາ​ເບິ່ງ​ຕົວ​ຢ່າງ​ການ​ຈັດການ​ໄຟ​ລ​ເມື່ອ​ມີ​ການ​ຮ້ອງ​ຂໍ​ຈາກ client ມາ​ທີ່ server ຂອງ PHP ກັບ Node.js ທຽບ​ກັນ​ເບິ່ງ​ເພື່ອ​ໃຫ້​ເຂົ້າໃຈ​ຫລາຍ​ຂຶ້ນ​

* ກໍລະນີ PHP ເມື່ອ​ມີ​ການ​ຮ້ອງ​ຂໍ​ເຂົ້າ​ມາ​ມັນ​ຈະ​ເຮັດ​ດັ່ງ​ນີ້:

1. ສົ່ງ task ໄປ​ທີ່​ລະບົບ​ໄຟ​ຂອງ​ຄອມພິວເຕີ
2. ລໍຖ້າຈົນກະທັ້ງ​ລະບົບ​ໄຟ​ລ​ເປີດ ​ແລະ ​ອ່ານ​ໄຟ​ລ​ສຳເລັດ
3. ສົ່ງ​ເນື້ອ​ຫາ​ຂອງ​ໄຟ​​ກັບ​ມາ​ໃຫ້ client
4. ພ້ອມ​ສຳລັບ​ຮັບ​ຄຳ​ຮ້ອງ​ຂໍ​ຖັດ​ໄປ

* ກໍລະນີ Node.js ເມື່ອ​ມີ​ການ​ຮ້ອງ​ຂໍ​ເຂົ້າ​ມາ​ມັນ​ຈະ​ເຮັດ​ດັ່ງ​ນີ້:

1. ສົ່ງ task ໄປ​ທີ່​ລະບົບ​ໄຟ​ຂອງ​ຄອມພິວເຕີ
2. ພ້ອມ​ສຳລັບ​ຮັບ​ຄຳ​ຮ້ອງ​ຂໍ​ຖັດ​ໄປ
3. ພໍ​ລະບົບ​ໄຟ​ລ​​ເປີດ ​ແລະ ​ອ່ານ​ໄຟ​ລ​ສຳເລັດ server ຈະ​ສົ່ງ​ເນື້ອ​ຫາ​ຂອງ​ໄຟ​ກັບ​ມາ​ໃຫ້ client

ຈາກ​ຕັວ​ຢ່າງ​ດ້ານ​ເທິງ​ຈະ​ເຫັນ​ວ່າ Node.js ຈະ​ຕັດ​ຂັ້ນ​ຕອນ​ການ​ລໍຖ້າຖີ້ມ​ແລ້ວ​ໄປ​ເຮັດ​ຄຳ​ຮ້ອງ​ຖັດ​ໄປ​ເລີຍທີ່​ເປັນ​ແບບນີ້​ເພາະ Node.js ຈະ​ run ​ແບບ single-threaded ແລະ​ໃນ​ library ມາດຕະຖານ​ກໍ​ຈະ​ມີ​ເຊັດ​ຂອງ asynchronous I/O primitives ທີ່​ຊ່ວຍ​ປ້ອງ​ກັນ​ໂຄດ JavaScript ຈາກ​ການ blocking ເຮັດໃຫ້​ລະບົບ​ຄ່ອງແຄ້ວ ແລະ ​ມີ​ປະ​ສິດ​ທິ​ພາບ​ຫຼາຍ​ຂຶ້ນ.

|  |  |
| --- | --- |
| **Synchronous vs Asynchronous** | |
| **Synchronous**ຄື​ການ​ run​ໂຄດ​ຕາມ​ລຳ​ດັບ​ທີ່​ເຮົາ​ຂຽນ​ໄວ້ ເຊັ່ນ  alert(1);  alert(2);  alert(3); ຜົນ​ລັບ​​ທີ່​ໄດ້​ຄື​ໜ້າ​ຈໍ​ຈະ​ສະແດງ​ຜົນ 1 ຈາກ​ນັ້ນ​ຈຶ່ງ​ສະແດງ​ຜົນ 2 ແລ້ວ​ສຸດ​ທ້າຍ​ຈຶ່ງ​ສະແດງ​ຜົນ 3 ຕາມ​ລຳ​ດັບ | **Asynchronous**ຄື​ການ​ຣັນໂຄ​ດ​ທີ່​ບໍ່​ຈຳ​ເປັນ​ຕ້ອງ​ເປັນ​ໄປ​ຕາມ​ລຳ​ດັບ​ທີ່​ເຮົາ​ຂຽນ​ໄວ້ ເຊັ່ນ:  alert(1);  set Time out(() => alert(2), 0);  alert(3); ໃນ​ຕົວ​ຢ່າງ​ນີ້​ໂຄດ alert(2) ໃຊ້​ເວລາ​ດຳ​ເນີນ​ການ​ດົນ​ກວ່າ ເຮັດໃຫ້​ຜົນ​ລັບ​​ທີ່​ໄດ້​ຄື​ໜ້າ​ຈໍ​ຈະ​ສະແດງ​ຜົນ 1 ຈາກ​ນັ້ນ​ຈຶ່ງ​ສະແດງ​ຜົນ 3 ແລ້ວ​ສຸດ​ທ້າຍ​ຈຶ່ງ​ສະແດງ​ຜົນ 2 |
| **Blocking vs Non-blocking** | |
| **Blocking**ໝາຍ​ເຖິງ​ການ​ທີ່​ເຮົາ​ບໍ່​ສາມາດ​ດຳ​ເນີນ​ການ​ຕໍ່ໄປ​ໄດ້​ຈົນ​ກວ່າ​ຕົວ​ດຳ​ເນີນ​ການ​ທີ່​ກຳ​ລັງ​ລັນ​ຢູ່​ຈະ​ສຳເລັດ​ເສຍ​ກ່ອນ ເຊັ່ນ:  alert(1);  var value = local Storage .get Item('foo');  alert(2); ຄຳ​ສັ່ງ local Storage ຈະ​ເປັນ​ຕົວ blocking ເຮັດໃຫ້​ບໍ່​ສາມາດ​ເຮັດ​ຄຳ​ສັ່ງ alert(2) ໄດ້​ຈົນ​ກວ່າ​ມັນ​ຈະ​ດຳ​ເນີນ​ການ​ສຳເລັດ ດັ່ງ​ນັ້ນ​ຜົນ​ລັບ​​ທີ່​ໄດ້​ຄື​ໜ້າ​ຈໍ​ຈະ​ສະແດງ​ຜົນ 1 ຈາກ​ນັ້ນ​ຈຶ່ງ​ລໍຖ້າຈົນກວ່າ​ຄຳ​ສັ່ງ local Storage ຈະ​ສຳເລັດ​ຈຶ່ງ​ສະແດງ​ຜົນ 2 | **Non-blocking**ໝາຍ​ເຖິງ​ການ​ໂຕທີ່​ດຳ​ເນີນ​ການ​ສາມາດ​ເຮັດ​ຄຳ​ສັ່ງ​ຖັດ​ໄປ​ໄດ້​ເລີຍ​ໂດຍ​ບໍ່​ຕ້ອງ​ລໍຖ້າ​ໃຫ້​ຄຳ​ສັ່ງ​ເດີມ​ເຮັດ​ສຳເລັດ​ກ່ອນເຊັ່ນ:  alert(1);  fetch('example.com').then(() => alert(2));  alert(3); ໃນ​ຕົວ​ຢ່າງ​ນີ້ ຄຳ​ສັ່ງ fetch ເປັນ non-blocking operation ດັ່ງ​ນັ້ນ​ຜົນ​ລັບ​ທີ່​ໄດ້​ຄື​ໜ້າ​ຈໍ​ຈະ​ສະແດງຜົນ 1 ຈາກ​ນັ້ນ​ຈຶ່ງ​ສະແດງ​ຜົນ 3 ແລ້ວ​ພໍ​ຄຳ​ສັ່ງ fetch ສຳເລັດ​ກໍ​ສະແດງ​ຜົນ 2 |

* ​**ປະຫວັດ Node.js**

ແຕ່​ເດີມ​ພາສາ JavaScript ຖືກ​ພັດທະນາ​ມາ​ເພື່ອ​ໃຊ້​ສຳລັບ Browserທີ່​ຊື່ Netscape Navigator ໃນ​ປີ 1995 ​ໃນ​ຕອນ​ນັ້ນ Netscape ຕັ້ງ​ໃຈ​ຈະ​ຂາຍWeb Server ທີ່​ມີ environment ຊື່ Netscape Live Wire ຊຶ່ງ​ສາມາດ​ສ້າງ dynamic page ໂດຍ​ໃຊ້ JavaScript ທາງ​ຝັ່ງ server ດ້ວຍແຕ່​ໜ້າ​ເສຍ​ດາຍ​ທີ່ Netscape Live Wireບໍ່​ປະ​ສົບ​ຄວາມ​ສຳ​ເລັດ ແລະ ​ການ​ໃຊ້JavaScriptທາງ​ຝັ່ງserver ກໍ​ບໍ່​ໄດ້ຮັບ​ຄວາມ​ນິຍົມ​ເລີຍ​ຈົນກະທັ້ງ Node.js ຖືກ​ກຳ​ເນີດ​ຂຶ້ນ​ມາ.

ສິ່ງ​ທີ່​ເຮັດໃຫ້ Node.js ເປັນ​ທີ່​ນິຍົມ​ຂຶ້ນ​ມາ​ຄື​ການ​ທີ່​ມັນ​ມາ​ໃນ​ຊ່ວງ​ເວລາ​ທີ່​ເໝາະ​ສົມ ເມື່ອ​ທຽບ​ກັບ JavaScript ທີ່​ເກີດ​ມາ​ຕັ້ງ​ແຕ່​ປີ 1995 ແລ້ວ Node.js ຫາກໍ່​ເກີດ​ມາ​ເມື່ອ​ປີ 2009 ເທົ່າ​ນັ້ນຕ້ອງ​ຂອບໃຈ "Web 2.0" applications (ເຊັ່ນ Flickr, Gmail ແລະ ອື່ນໆໃຫ້ສະເເດງທີ່ໜ້າເວັບ​ສະໄໝ​ໃໝ່​ຄວນ​ໜ້າ​ຕາ​ເປັນ​ຢ່າງໃດ

ບໍ່ດົນ​ກ່ອນ​ທີ່ Node.js ຈະ​ເກີດ ນັກ​ພັດທະນາ​ເບື້ອງ​ຫຼັງ browser ຊື່​ດັ່ງ​ທັງ​ຫຼາຍ​ແຂ່ງ​ຂັນ​ກັນ​ເຮັດວຽກ​ຢ່າງ​ໜັກ​ເພື່ອຈະ​ໃຊ້ JavaScript ໃຫ້​ໄດ້​ດີ​ທີ່ສຸດ ​ແລະ​ ຫາ​ທາງ​ເຮັດໃຫ້ JavaScript ສາມາດ​ລັນ​ໄດ້​ໄວ​ຫລາຍ​ຂຶ້ນເພື່ອ​ໃຫ້​ຜູ້​ໃຊ້​ງານ​ໄດ້ຮັບ​ປະ​ສິດ​ທິ​ພາບ​ທີ່​ດີ​ທີ່ສຸດຊຶ່ງ​ຜົນ​ຈາກ​ການ​ແຂ່ງ​ຂັນ​ນີ້​ເຮັດໃຫ້​ເກີດ​ການ​ພັດທະນາ Chrome V8 (open-source JavaScript engine ຂອງ The Chromium Project) ຂຶ້ນ​ມາ ແລະ Node.js ກໍ​ໃຊ້ engine ນີ້​​ເອງ

ແຕ່​ການ​ທີ່ Node.js ເປັນ​ທີ່​ນິຍົມ​ຂຶ້ນ​ມາ​ບໍ່​ແມ່ນ​ແຕ່ວ່າ​ມັນ​ມາ​ຖືກ​ທີ່​ຖືກ​ເວລາ​ເທົ່າ​ນັ້ນ​ແຕ່​ເພາະ​ມັນ​ໄດ້​ສະແດງ​ໃຫ້​ເຫັນ​ແລ້ວ​ວ່າ ການ​ອອກ​ແບບ ​ແລະ ​ແນວ​ຄິດ​ຂອງ​ມັນ​ຊ່ວຍ​ນັກ​ພັດທະນາ​ທັງ​ຫຼາຍ​ໃຫ້​ສາ

ມາດ​ໃຊ້ JavaScript ທາງ​ຝັ່ງ server ໄດ້​ງ່າຍ​ຂຶ້ນ​ຫລາຍ​ອີກດ້ວຍ​.

* + 1. **ຄວາມຮູ້ກ່ຽວກັບ React**

Reactເປັນເທັກໂນໂລຢີໜຶ່ງທີ່ມາແຮງຫຼາຍທັງໃນໄທແລະຕ່າງປະເທດໂດຍສ້າງມາຈາກພື້ນຖານແນວຄວາມຄິດແບບ MVC (Model View Controller) ຊຶ່ງໝາຍຄວາມວ່າ React ມີຫນ້າທີ່ຈັດການກັບ Model ຫລື View ແຕ່ສ່ວນໃຫຍ່ຈະເປັນ View ກ່ອນໜ້ານັ້ນເວລາຈະຂຽນໜ້າເວັບ ເຮົາກໍຈະເຮັດຜ່ານ HTML ມີການໃຊ້ CSS ໃນການປັບປ່ຽນໜ້າຕາຂອງ UI ແຕ່​ໃນ React ຖ້າ​ຈະ​ສ້າງ​ໜ້າ​ເວັບ​ຂຶ້ນ​ມາ​ໄດ້​ນັ້ນ ເຮົາ​ຈະ​ໃຊ້​ເປັນ [Component](https://www.mindphp.com/%E0%B8%9A%E0%B8%97%E0%B9%80%E0%B8%A3%E0%B8%B5%E0%B8%A2%E0%B8%99%E0%B8%AD%E0%B8%AD%E0%B8%99%E0%B9%84%E0%B8%A5%E0%B8%99%E0%B9%8C/tutorial-reactjs/4864-component-and-props.html) ປຽບໄດ້​ວ່າ Component ເປັນ Block ສ່ວນ​ຍ່ອຍ​ຂອງ​ເວັບ​ເຮົາ​ທີ່​ຈະ​ສ້າງ​ອອກ​ມາ ໂດຍ​ເຮົາ​ຈະ​ຂຽນ​ໂຄດພາສາ [JavaScript](https://www.mindphp.com/%E0%B8%9A%E0%B8%97%E0%B9%80%E0%B8%A3%E0%B8%B5%E0%B8%A2%E0%B8%99%E0%B8%AD%E0%B8%AD%E0%B8%99%E0%B9%84%E0%B8%A5%E0%B8%99%E0%B9%8C/%E0%B8%AA%E0%B8%AD%E0%B8%99-javascript/2724-%E0%B8%9A%E0%B8%97%E0%B8%97%E0%B8%B5%E0%B9%88-1-javascript-introduction.html) เພື່​ອະທິບາຍ​ໜ້າ​ຕາຂອງ block ນັ້ນ​ໆ​ມາ,ເວລາ​ສະແດງ​​ໜ້າ Page ເຮົາ​ຈະ​ເອີ້ນ Component ອອກ​ມາ​ສະແດງ​ຜົນໜ້າ​ຕາ​ຂອງ​ໂຄດ​ດັ່ງຕົວຢ່າງ​ນີ້:

ReactDOM.render(

<h1>Hello, world!</h1>,

document.getElementById('root')

);



**ຮູບທີ 11** React.js

* **ສະຫລຸບ​ງ່າຍໆ ມີ​ດັ່ງ​ນີ້**
* React ເປັນ JavaScript Libraly ສ້າງ​ ແລະ​ ພັດທະນາ​ຂຶ້ນ​ໂດຍ Facebook ຂຽນ​ໄດ້​ແຕ່ UI

ເທົ່າ​ນັ້ນ ແລະ ​ເປີດ​ໃຫ້​ໃຊ້​ຟ​ຣີ

* React ໃຊ້​ໂຄດ HTML, CSS ແລະ JavaScript
* React ມີ 3 ຄອນ​ເຊບ​ທີ່​ເຮົາ​ຕ້ອງ​ຮຽນ​ຮູ້ ຄື Component, State ແລະ Props
* **ຂໍ້​ດີ​ແລະ​ຂໍ້​ເສຍ​ຂອງ React**

**ກ. ຂໍ້​ດີ**

* Component ເຂົ້າໃຈ​ງ່າຍ ເຮົາ​ສາມາດ​ຮຽນ​ຮູ້​ໄດ້​ດ້ວຍ​ຕົວ​ເອງ
* Tool ຫຼາຍ ພຽງ React ຢ່າງ​ດຽວ​ກໍ່ສາມາດຂຽນເວັບ​ໄດ້​ທັງ​ເວັບ​ແລ້ວ ໂດຍ​ບໍ່​ຕ້ອງ​ຫາ Tool ເພີ່ມເຕີມ ແລະ ​ຍັງມີ Tool ພັດທະນາ​ອອກ​ມາ​ຢູ່​ເລື້ອຍໆສາມາດ​ເຮັດ App ໄດ້ React ມີ​ເຄື່ອງ​ມື​ໜຶ່ງ​ທີ່​ຊື່​ວ່າ React Native ເປັນ​ການ​ຂຽນ JavaScript ແລ້ວ​ແປງ​ເປັນ App ແບບ Native ໄດ້​ທັງ​ເທິງ Android ແລະ IOS

**ຂ. ຂໍ້​ເສຍ**

* ຕ້ອງ​ມີ​ພື້ນ​ຖານ​ໃນ JavaScript ໃນ​ລະ​ດັບ​ໜຶ່ງ ຄົົນທີ່​ສົນ​ໃຈ​ຮຽນ​ກໍ​ຈະ​ລຳ​ບາກ​ນ້ອຍ​ໜຶ່ງ​ຖ້າ​ບໍ່​ໄດ້​ມີ​ພື້ນ​ຖານ JavaScript ອາດຈະ​ຕ້ອງ​ໃຊ້​ເວລານ້ອຍໜື່ງ

- Docmentation ອ່ານ​ຍາກ React ມີ Documentation ທີ່​ຍັງ​ບໍ່​​ຄ່ອຍ​ດີ ໃນ​ອະນາຄົດ​ອາດຈະມີ​ ການ​ພັດທະນາ​ອີກ.

* + 1. **ຄວາມຮູ້ກ່ຽວກັບໂປຣແກຣມ Microsoft Office 2016**

Microsoft word ແມ່ນໂປຣແກຣມປະມວນຜົນ ເຊີ່ງອອກແບບມາເພື່ອຊ່ວຍໃຫ້ສ້າງເອກະສານທີ່ມີຄຸນນະພາບໃນລະດັບມືອາຊີບ ເຄື່ອງມືການຈັດຮູບແບບເອກະສານທີ່ດີທີ່ສຸດຂອງ word ຈະເຮັດໃຫ້ສາມາດຈັດລະບຽບ ແລະ ຂຽນເອກະສານຂອງທ່ານໄດ້ຢ່າງມີປະສິດທິພາບຫຼາຍຂື້ນ word ຍັງມີເຄື່ອງມືການແກ້ໄຂ ແລະ ກວດສອບຄຳຜິດທີ່ມີປະສິດທິພາບ ສາມາດເຮັດວຽກຮ່ວມກັບຜູ້ອື່ນໄດ້ຢ່າງງ່າຍດາຍ.

1. ປະໂຫຍດຂອງໂປຣແກຣມ Microsoft Word Office 2016

* ມີລະບົບປະຕິບັດຕ່າງໆທີ່ຊ່ວຍໃນການເຮັດວຽກໃຫ້ສະດວກສະບາຍຂື້ນເຊັ່ນ: ການກວດຄຳສະກົດ, ການກວດສອບໄວຍະກອນ, ການໃສ່ຂໍ້ຄວາມອັດຕະໂນມັດເປັນຕົ້ນ.
* ສາມາດໃຊ້ສ້າງຈົດໝາຍໄດ້ຢ່າງວ່ອງໄວໂດຍສາມາດກຳນົດໃຫ້ຜູ້ວິເສດ(Wizard)ໃນ word ສ້າງແບບຟອມຂອງຈົດໝາຍໄດ້ຫຼາຍຮູບແບບຕາມທີ່ຕ້ອງການ.
* ສາມາດໃຊ້ word ສ້າງຕາຕະລາງທີ່ສະຫຼັບຊັບຊ້ອນໄດ້.
* ປັບປຸງເອກະສານໄດ້ງ່າຍ ແລະ ວ່ອງໄວສາມາດຕົກແຕ່ງເອກະສານ ຫຼື ເພື່ອຄວາມສະດວກ
* ຈະໃຫ້ word ສາມາດແຊກຮູບພາບ, ກຣາຟິກ ຫຼື ຜັງອົງກອນລົງໃນເອກະສານໄດ້.
* Word ປັບແຕ່ງໃຫ້ກໍໄດ້ ໂດຍສາມາດເປັນຜູ້ກຳນົດຮູບແບບຂອງເອກະສານເອງ.
* ຄວາມສາມາດໃນການເຊື່ອມຕໍ່ໂປຣແກຣມອື່ນໆ ໃນຊຸດໂປຣແກມ Microsoft office ສາມາດ

ໂອນຍ້າຍຂໍ້ມູນຕ່າງໆ ລະຫ່ວາງໂປຣມແກມໄດ້.

* ສ້າງເອກະສານໃຫ້ໃຊ້ວຽກໃນອິນເຕີເນັດໄດ້ຢ່າງງ່າຍດາຍ.

1. ຄວາມສາມາດຂອງ Microsoft word

* ສາມາດພິມ ແລະ ແກ້ໄຂເອກະສານ.
* ສາມາດລຶບ, ຍ້າຍ ແລະ ຮ່າງຂໍ້ຄວາມ.
* ສາມາດພິມຕົວອັກສອນປະເພດຕ່າງໆ.
* ສາມາດຂະຫຍາຍຕົວອັກສອນ.
* ສາມາດຈັດຕົວອັກສອນໃຫ້ເປັນຕົວໜາ.,ຕົວຫງ່ຽງ ແລະ ຂີດກ້ອງໄດ້.
* ສາມາດໃສ່ເຄື່ອງໝາຍ ແລະ ຕົວເລກລຳດັບຕາມຫົວຂໍ້.
* ສາມາດແບ່ງເສົາໄດ້.
* ສາມາດກວດການສະກົດ ແລະ ແກ້ໄຂໃຫ້ຖຶກຕ້ອງ.
* ສາມາດຄົ້ນຫາ ແລະ ປ່ຽນແປງຂໍ້ຄວາມທີ່ພິມຜິດ.
* ສາມາດຈັດຂໍ້ຄວາມເອກະສານໃຫ້ພິມໄປທາງຊ້າຍ, ທາງຂວາ ແລະ ລະຫວ່າງກາງ.
* ສາມາດໃສ່ຮູບພາບໃນເອກະສານ.
* ສາມາດຕົກແຕ່ງໂຕອັກສອນ,ພິມຕາຕະລາງ.
  + 1. **ຄວາມຮູ້ກ່ຽວກັບໂປຣແກຣມ Microsoft Visio 2016**

ໂປຣແກຣມ Microsoft Visio ເປັນເຄື່ອງມືເສີມການເຮັດວຽກຂອງ Microsoft Office ໃນການສ້າງແຜນວາດ (Diagram) ປະເພດຕ່າງໆ, ເຊີ່ງເຮັດໄດ້ງ່າຍດາຍ,ສະດວກ ແລະ ວ່ອງໄວ. ເປັນທີ່ນິຍົມໃຊ້, ມີປະໂຫຍດຫຼາຍສຳລັບການຈັດເອກະສານ, ລວມໄປເຖິງແຜນວາດການອອກແບບຂັ້ນຕອນເຮັດການວິເຄາະ ແລະ ອອກກແບບລະບົບຕ່າງໆ.

ສຳລັບໂປຣແກຣມ Microsoft Visio ມີໃຫ້ເລືອກຢູ່ 2 ປະເພດຄື:

1. Microsoft Visio Standard ເໝາະກັບວຽກດ້ານທຸລະກິດເຊັ່ນ: ຜູ້ບໍລິຫານໂຄງການ, ນັກການຕະຫຼາດ, ພະນັກງານຝ່າຍຊັບພະຍາກອນມະນຸດ ແລະ ທິມງານ ມີໜ໊າທີ່ເບີ່ງແຍງການດຳເນີນງານ ເພື່ອຊ່ວຍໃນການເບີ່ງແຜນວາດ ແລະ ຂ່າວສານ.
2. Microsoft Visio Professional ເໝາະສຳລັບມືອາຊີບທາງດ້ານເຕັກນິກ, ພະນັກງານໄອທີ, ນັກພັດທະນາ ແລະ ວິສາວະກອນທີ່ຊ່ວຍໃນການອອກແບບຂໍ້ມູນລະບົບ ເພື່ອໃຊ້ໃນການເຮັດຕົ້ນແບບ. Microsoft Visio Professional ເປັນໂປຣແກຣມ ທີ່ຖືກສ້າງຂື້ນມາເພື່ອຊ່ວຍໃນການສ້າງ Flow chart ຫຼື Diagram ຂອງວຽກໃນສາຂາຕ່າງໆໃຫ້ເຮັດວຽກໄດ້ງ່າຍຂື້ນ. ລັກສະນະຢ່າງໜື່ງໃນການສ້າງFlow chart ໃນMicrosoft Visio Professional ກໍຄືມີຮູບແບບDiagram ພື້ນຖານຕ່າງໆ ຈັດກຽມໄວ້ໃຫ້, ເຊີ່ງງ່າຍໃນການອອກແບບ ແລະ ໃຊ້ວຽກ.

ຈຸດດີ Microsoft Visio ຄື:ຊອກຫາຂໍ້ຜິດພາດໄດ້ຢ່າງສະດວກ ແລະ ເຂົ້າໃຈງ່າຍໃນການ

ສະແດງທິດທາງການໄຫຼຂໍ້ມູນ.

* + 1. **ຄວາມໝາຍ ແລະ ຄວາມສຳຄັນຂອງການວິເຄາະລະບົບ**

2.1.10.1 ຄວາມໝາຍຂອງການວິເຄາະລະບົບ.

ການວິເຄາະລະບົບ ແມ່ນການສືກສາບັນທີ່ເກີດຂື້ນກັບລະບົບ ເພື່ອມາອອກແບບການເຮັດວຽກໃໝ່ຂອງລະບົບ, ນອກຈາກນີ້ຍັງເພື່ອແນໃສ່ປັບປຸງ ແລະ ແກ້ໄຂລະບົບການເຮັດວຽກແບບເດີມໃຫ້ດີຂື້ນກ່ວາເກົ່າ.

2.1.10.2 ຄວາມສຳຄັນຂອງການວິເຄາະລະບົບ.

* ເພື່ອໃຊ້ເປັນເຄື່ອງມືທີ່ເຮັດໃຫ້ຮູ້ເຖິງຄວາມຕ້ອງການຂອງຜູ້ໃຊ້ຢ່າງຊັດເຈນ.
* ເພື່ອໃຊ້ໃນການກວດສອບຄວາມເປັນໄປໄດ້ຂອງການວິເຄາະ ແລະ ອອກແບບລະບົບໃຫ້ໄດ້ຊັດເຈນເຖີງຜົນກະທົບຂອງລະບົບທີ່ໄດ້ວິເຄາະ ແລະ ອອກແບບ, ຊອກຫາທາງໃໝ່ເພື່ອແກ້ໄຂຜົນກະທົບນັ້ນ.
  + 1. **ແຜນວາດຄວາມສຳພັນຂອງ Entity (ER Diagram)**

ການອອກແບບການສ້າງຕາຕະລາງຖານຂໍ້ມູນແບບຈໍາລອງ ER ແບ່ງອອກເປັນ 2 ຂັ້ນຕອນຫຼັກ ຄື: ຂັນຕອນທໍາອິດເປັນການສ້າງແບບຈໍາລອງ ER ຂັ້ນຕອນນີ້ຈະກ່າວເຖີງຄວາມໝາຍຂອງສັນຍາລັກຕ່າງໆທີ່ໃຊ້ໃນແບບຈໍາລອງ ER, ຂັ້ນຕອນທີ່ 2 ແມ່ນການແປງແບບຈໍາລອງ ER ໃຫ້ເປັນຕາຕະລາງຂໍ້ມູນເພື່ອໃຊ້ເປັນຕົວຈັດການຂໍ້ມູນ ເມື່ອເຂົ້າໃຈໃນທັງສອງແບບນີ້ແລ້ວສາມາດອອກແບບຖານຂໍ້ມູນ ແລະ ສ້າງຖານຂໍ້ມູນໃນແບບຈໍາລອງ ER ຂຶ້ນມາໃຊ້ງານດ້ວຍຕົນເອງ.

### ສັນຍາລັກທີ່ໃຊ້ໃນແບບຈໍາລອງ ER

ການອອກແບບຈໍາລອງຖານຂໍ້ມູນ ER ນັ້ນກ່ອນອື່ນຕ້ອງຮູ້ຈັກຄວາມໝາຍສັນຍາລັກຕ່າງໆທີ່ໃຊ້ໃນການອອກແບບຈໍາລອງ ER ໄດ້ຢ່າງຖືກຕ້ອງດັ່ງນັ້ນໃນຫົວຂໍ້ນີ້ຈະເວົ້າເຖິງຄວາມໝາຍ ແລະ ການໃຊ້ງານສັນຍາລັກຕ່າງໆຂອງ ER.

* 1. ອັນຕີຕີ້ (Entity)

ເອັນຕີຕີ້ຄືວັດຖຸທີ່ເຮົາສົນໃຈເຊິ່ງອາດເປັນໄດ້ທັງບຸກຄົນ,ສະຖານທີ່,ວັດຖຸ,ເຫດການ ຫຼື ແນວຄິດທີ່ກໍ່ໃຫ້ເກີດກຸ່ມຂອງຂໍ້ມູນທີ່ຕ້ອງການເອັນຕີຕີ້ແບ່ງອອກເປັນ 2 ປະເພດຄື:

* 1. Strong Entity:

ເປັນເອັນຕີຕີ້ທີ່ເກີດຂຶ້ນດ້ວຍຕົນເອງເປັນອິດສະຫຼະບໍ່ຂຶ້ນກັບເອັນຕີຕີໃດສັນຍາລັກທີ່ໃຊ້ຄືຮູບສີ່ຫຼ່ຽມ ແລະ ສາມາດເອີ້ນStrong Entity ໄດ້ອີກຊື່ໜື່ງວ່າ Regular Entity.

**CUSTOMER**

**STOCK**

**ຮູບທີ 3** ຮູບStrong Entity

* 1. Weak Entity:

ເອັນຕີຕີ້ຊະນິດນີ້ຈະຂຶ້ນກັບເອັນຕີຕີ້ຊະນິດອື່ນໆບໍ່ສາມາດເກີດຂຶ້ນໄດ້ຕາມລໍາພັງ ແລະ ຈະຖືກລົບເມື່ອເອັນຕີຕີ້ຫຼັກຖືກລົບອອກ ສັນຍາລັກທີ່ໃຊ້ຄືຮູບສີ່ຫຼ່ຽມຊ້ອນກັນ.

**CUSTOMER**

**STOCK**

**ຮູບທີ 4** ຮູບWeak Entity

* 1. ແອັດທີຣບີວ(Attribute)

ແອັດທີຣບີວຄືຄຸນສົມບັດຂອງສົມບັດຂອງເອັນຕີຕີ້ ສັນຍາລັກຂອງແອັດທີຣບີວຈະເປັນຮູບ ວົງຮີແອັດທີຣບີວໃດທີ່ຖືກໃຊ້ເປັນຄີຫຼັກຈະຖືກຂີດເສັ້ນກ້ອງກໍາກັບໄວ້.

emName

Position

emNo

**EMPLOYEE**

**ຮູບທີ 5** ຮູບAttribute

* 1. ສໍາພັນ (Relation)

ຄວາມສໍາພັນໃນທີ່ນີ້ໝາຍເຖິງຄວາມສໍາພັນລະຫວ່າງເອັນຕີຕີ້ໂດຍແຕ່ລະຄວາມສໍາພັນຄວນມີຊື່ລະບຸໄວ້ເພື່ອໃຊ້ອະທິບາຍເຊິ່ງປົກກະຕິຈະໃຊ້ສັນຍາລັກຮູບດອກຈັນທີ່ພາຍໃນລະບຸຄໍາກິລິຍາໄວ້ເພື່ອອະທິບາຍຄວາມສໍາພັນ.

SUBJECT

teaches

LUCTURE

M

1

**ຮູບທີ 6** ຮູບAttribute

* + 1. **ຄວາມໝາຍ ແລະ ສັນຍະລັກຂອງ Flowchart**

|  |  |  |
| --- | --- | --- |
| **ຊື່** | **ສັນຍາລັກ** | **ຄວາມຫມາຍ** |
| Boundary Or External Entity |  | ຂອບເຂດຫມາຍເຖິງພາກສ່ວນທີ່ກ່ຽວຂ້ອງເກັບລະບົບເຊິ່ງລະບົບບໍ່ສາມາດຄວບຄຸມໄດ້ |
| Process |  | ປະມວນຜົນຫຼືຫນ້າວຽກທີ່ເຮັດໃນໂຄງການນັ້ນໆ |
| Data Store |  | ບ່ອນຈັດເກັບຂໍ້ມູນ |
| Data Flow |  | ການໄຫຼຂອງຂໍ້ມູນ |
| Real-Time Link |  | ການເຊື່ອມໂຍງແບບໄກທີ່ມີການຕອບກັບແບບທັນທີທັນໃດ |

**ຕາຕະລາງທີ 2** ສະແດງສັນຍາລັກ Data Flow Diagram

1. **ຈຸດປະສົງຂອງ DFD**

* ເປັນແຜນພາບທີ່ສະຫຼຸບລວມຂໍ້ມູນທັງໝົດໄດ້ຈາກການວິເຄາະໃນລັກສະນະຂອງຮູບແບບທີ່ ເປັນໂຄງສ້າງ.
* ເປັນຂໍ້ຕົກລົງຮ່ວມກັນລະຫວ່າງນັກວິເຄາະລະບົບ ແລະ ຜູ້ຊົມໃຊ້.
* ເປັນແຜນພາບທີ່ໃຊ້ໃນການພັດທະນາຕໍ່ໃນຂັ້ນຕອນຂອງການອອກແບບ.
* ຮູ້ທີ່ໄປທີ່ມາຂອງຂໍ້ມູນທີ່ໄຫຼໃນຂະບວນການຕ່າງໆ.

1. **ກົດຂອງ Process**

* ຕ້ອງບໍ່ມີຂໍ້ມູນເຂົ້າພຽງຢ່າງດຽວ
* ຕ້ອງບໍ່ມີຂໍ້ມູນອອກພຽງຢ່າງດຽວ
* ຂໍ້ມູນທີ່ສົ່ງເຂົ້າຕ້ອງພຽງພໍໃນການສ້າງຖານຂໍ້ມູນທີ່ສົ່ງອອກ
* ການຕັ້ງຊື່ Process ຕ້ອງໃຊ້ຄໍາກິລິຍາ(Verb) ເຊັ່ນ: ບັນທຶກຂໍ້ມູນ,ໃບບິນ,ກວດສອບຂໍ້ມູນລູກຄ້າ,ຈໍານວນເງິນເດືອນ.

### ປຽບທຽບແຜນວາດການໄຫຼຂໍ້ມູນທີ່ບໍ່ຖືກຕ້ອງ ແລະ ຖືກຕ້ອງ

1. **Process:** 
   * ເມື່ອມີຂໍ້ມູນເຂົ້າໄປທີ່ Processກໍ່ຕ້ອງມີຂໍ້ມູນ ຫຼື ຜົນຮັບອອກມາຈາກ Processເຊັ່ນກັນຈະເປັນໄປບໍ່ໄດ້ທີ່ມີສະເພາະຂໍ້ມູນເຂົາຢ່າງດຽວ.
2. Data store:

* ຂໍ້ມູນຈະໄຫຼຈາກ Data store ໜຶ່ງໄປຫາ Data store ໜຶ່ງໂດຍກົງບໍ່ໄດ້ຈະຕ້ອງຜ່ານProcessເທົ່ານັ້ນ
* ຂໍ້ມູນທີ່ສົ່ງຜ່ານ External entity ບໍ່ສາມາດໄຫຼເຂົ້າໄປ Data store ໂດຍກົງໄດ້ຈະຕ້ອງໃຊ້ Processເປັນຕົວກາງໃນການເຊື່ອມໂຍງເພື່ອຈັດເກັບຂໍ້ມູນໃນ Data store
* ຂໍ້ມູນທີ່ໄຫຼຜ່ານຈາກ Data storeບໍ່ສາມາດເຊື່ອໂຍງເຂົ້າກັບ External entityໄດ້ໂດຍກົງຈະຕ້ອງຜ່ານ Process ເທົ່ານັ້ນ.

1. External entity:

* External entityບໍ່ສາມາດເຊື່ອມໂຍງເຂົ້າຫາກັນໄດ້ຈະຕ້ອງໃຊ້ Processເປັນຕົວກາງເພື່ອສົ່ງຜ່ານ ແລະ ຊື່ຂອງ External entityຈະໃຊ້ຄໍານາມເທົ່ານັ້ນ.

1. Data flow:

* ການໄຫຼຂໍ້ມູນທີ່ມີຫົວຊີ້ໄປທີ່ Process ໝາຍເຖິງ Process ມີການອ່ານ ຫຼື ການດືງຂໍ້ມູນຈາກ Data store ມາໃຊ້ວຽກ
* ການໄຫຼຂໍ້ມູນຈາກ Process ທີ່ມີຫົວລູກສອນຊີ້ໄປຍັງ Data store ໝາຍເຖິງການ Update ຫຼື ການເພີ່ມຂໍ້ມູນລົງໄປທີ່ Data store

**ຕາຕະລາງທີ 10** ຮູບການປຽບທຽບແຜນວາດການໄຫຼຂໍ້ມູນທີ່ບໍ່ຖືກຕ້ອງ ແລະ ຖືກຕ້ອງ

|  |  |
| --- | --- |
| **ອະນຸຍາດ** | **ບໍ່ອະນຸຍາດ** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

* ການໄຫຼຂໍ້ມູນທີ່ມີຫົວລູກສອນທັງສອງດ້ານທີ່ເຊື່ອມໂຍງລະຫວ່າງ Process ກັບ Data store ໝາຍເຖິງມີການດືງຂໍ້ມູນຈາກ Data store ມາປັບປຸງ ແລະ ມີການ Update ຂໍ້ມູນລົງໄປໃນ Data store
* ການໄຫຼຂໍ້ມູນບໍ່ສາມາດຢ້ອນກັບໄປຍັງ Process ເດີມໄດ້ຢ່າງນ້ອຍຕ້ອງເຊື່ອມໂຍງຜ່ານProcess ໜື່ງເພື່ອສົ່ງຜ່ານຢ້ອນກັບມາຍັງ Process ເດີມ ແລະ ຊື່ທີ່ລະບຸໃນການໄຫຼຂໍ້ມູນຈະໃຊ້ຄໍານາມ.

# **ບົດທີ 3**

# **ວິທີດຳເນີນການຄົ້ນຄວ້າ**

## ວິທີສຶກສາ ແລະ ຄົ້ນຄວ້າ

ຂັ້ນຕອນວິທີການດຳເນີນການພັດທະນາລະບົບໃນຄັ້ງນີ້ແມ່ນອີງໃສ່ລັກສະນແບບຈຳລອງຂອງ Adapted Waterfall Model ເປັນແນວທາງໃນການວິເຄາະ ແລະ ອອແບບລະບົບ ເຊິ່ງປະກອບດ້ວຍໜ້າວຽກດັ່ງນີ້:

ພັດທະນາ

ທົດສອບ

ສ້າງເອກະສານ

ວາງແຜນ

ວິເຄາະ

ອອກເບບ

ແຜນວາດທີ່1: ຂັ້ນຕອນວິທີດຳເນີນການຂອງໂຄງການ

* + 1. **ໄລ່ຍະທີ 1 ການວາງແຜນໂຄງການ (Project planning phase)**

ການວາງແຜນໂຄງການຈັດເປັນຂະບວນການພື້ນຖານ ໃນຄວາມເຂົ້າໃຈຢ່າງເລິກເຊິ່ງວ່າເປັນຫຍັງຕ້ອງສ້າງລະບົບໃໝ່ ທິມງານຕ້ອງພິຈາລະນາວ່າຈະຕ້ອງດຳເນີນງານຕໍ່ໄປແນວໃດ ກ່ຽວກັບຂະຂະບວນການສ້າງລະບົບໃໝ່ ກ່ອນອື່ນໝົດຕ້ອງມີຈຸດກຳເນີດຂອງລະບົບງານ ເຊິ່ງໂດຍປົກກະຕິແລ້ວຈຸດກຳເນີດຂອງລະບົບງານມັກເກີດຂຶ້ນຈາກຜູ້ໃຊ້ລະບົບເປັນຜູ້ທືີຕິດແທດກັບລະບົບໂດຍກົງເຮັດໃຫ້ມີຄວາມໃກ້ສິດກັບລະບົບງານທີ່ດຳເນີນຢູ່ຫຼາຍທີ່ສຸດ ເມື່ອຜູ້ໃຊ້ລະບົບມີຄວາມຕ້ອງການປັບປຸງ ລະບົບ ງານດັ່ງນັ້ນຈຶ່ງຖືເປັນຈຸດເລີ່ມຕົ້ນ ໃນບົດບາດຂອງນັກວິເຄາະລະບົບວ່າຈະຕ້ອງສຶກສາ ເຖິ່ງຂອບເຂດ ຂອງບັນຫາທີ່ຜູ້ໃຊ້ລະບົບກຳລັງປະສົບບັນຫາຢູ່ ແລະຈະດຳເນີນການແກ້ໄຂແນວໃດສຶກສາເຖິງຄວາມເປັນໄປໄດ້ວ່າລະບົບໃໝ່ທີ່ຈະພັດທະນາຂຶ້ນມານັ້ນມີຄວາມເປັນໄປໄດ້ ແລະ ຄຸ້ມຄ່າກັບການຈະລົງທຶນ

ຫຼືບໍ່.

ແນວໃດກໍ່ຕາມໄລຍະຂອງການວາງແຜນໂຄງການປົກກະຕິມັກຈະມີໄລຍະເວລາທີ່ສັ້ນ ແຕ່ກໍ່ຖືວ່າເປັນຂັ້ນຕອນທີ່ສຳຄັນທີ່ຈະໃຫ້ເກີດຜົນສຳເລັດໄດ້ ດັ່ງນັ້ນໃນໄລຍະຂອງການວາງແຜນໂຄງການຈຶ່ງໄດ້ອາໄສນັກວິເຄາະລະບົບທີ່ມີຄວາມຮູ້ ແລະ ປະສົບການສູງເນື່ອງຈາກວ່າຫາກນັກວິເຄາະລະບົບບໍ່ເຂົ້າໃຈເຖິ່ງບັນຫາອັນແທ້ຈິງທີ່ເກີດຂຶ້ນກໍ່ຈະບໍ່ສາມາດພັດທະນາລະບົບຂຶ້ນມາເພື່ອແກ້ໄຂບັນຫາໃຫ້ຖືກຈຸດໄດ້ ແລະ ມັກຈະມີໂຄງການພັດທະນາລະບົບຫຼາຍໂຄງການ ທີ່ຫຼັງຈາກໄດ້ດຳເນີນການພັດທະນາ ແລະ ນໍາມາໃຊ້ງານແລ້ວປະກົດວ່າບໍ່ສາມາດຕອບສະໜອງຄວາມຕ້ອງການຂອງຜູ້ໃຊ້ງານ ເຊິ່ງຖືວ່າເປັນເລື່ອງທີ່ກໍ່ໃຫ້ເກີດຄວາມສູນເສຍທັງທາງດ້ານການລົງທຶນ ແລະ ໄລຍະເວລາ.

ສະຫຼຸບລວມແລ້ວໄລຍະຂອງການວາງແຜນໂຄງການປະກອບມີກິດຈະກໍາຕ່າງໆຕໍ່ໄປນີ້:

- ກໍານົດບັນຫາ Problem Definition

- ສຶກສາຄວາມເປັນໄປໄດ້ຂອງໂຄງການ Feasibility study

- ສ້າງຕາຕະລາງກໍານົດເວລາໂຄງການ Project scheduling

- ຈັດຕັ້ງທິມງານໂຄງການ Staff the project

* + 1. **ໄລ່ຍະທີ 2 ການວິເຄາະ (Analysis Phase)**

ໄລຍະການວິເຄາະ ຈະຕ້ອງມີຄຳຕອບກ່ຽວກັບຄຳຖາມວ່າໃຜເປັນຜູ້ທີ່ໃຊ້ລະບົບແລະມີຫຍັງແດ່ທີ່ຈະຕ້ອງເຮັດໃນໄລຍະນີ້ນັກວິເຄາະລະບົບຈະຕ້ອງດຳເນີນການໃນຂັ້ນຕອນຂອງການວິເຄາະລະບົບງານປັດຈຸບັນ(Currentsystem)ເພື່ອນໍາມາພັດທະນາແນວຄວາມຄິດສໍາລັບລະບົບໃໝ່ (New System).

ຈຸດປະສົງຫຼັກຂອງການວິເຄາະຄືຈະຕ້ອງສຶກສາ ແລະ ສ້າງຄວາມເຂົ້າໃຈໃນຄວາມຕ້ອງການຕ່າງໆ ທີ່ໄດ້ລວບລວມມາດັ່ງນັ້ນການລວບລວມຄວາມຕ້ອງການ (Requirements Gathering) ຈຶ່ງຈັດເປັນງານສ່ວນພື້ນຖານຂອງການວິເຄາະລະບົບໂດຍຂໍ້ມູນຄວາມຕ້ອງການເຫຼົ່ານີ້ນັກວິເຄາະລະບົບຈະນຳມາວິເຄາະເພື່ອທີ່ຈະປະເມີນວ່າຄວນມີຫຍັງແນ່ທີ່ລະບົບໃໝ່ຕ້ອງດຳເນີນການ ແລະ ດ້ວຍເຫດນີ້ເອງການກຳນົດລາຍລະອຽດກ່ຽວກັບຄວາມຕ້ອງການຂອງຜູ້ໃຊ້ (User Requirements)ຈະທະວີຄວາມສຳຄັນຫຼາຍຂຶ້ນ ເປັນລຳດັບສຳລັບລະບົບທີ່ມີຄວາມສັບຊ້ອນສູງ ແລະ ນັກວິເຄາະຕ້ອງເອົາໃນໃສ່ກັບການລວບລວມຄວາມຕ້ອງການຈາກຜູ້ໃຊ້ ບໍ່ຄວນກຳນົດຄວາມຕ້ອງການຂຶ້ນເອງໂດຍໃຊ້ຄວາມຄິດສ່ວນຕົວຂອງຕົນເອງເປັນຫຼັກຫຼື່ປະເມີນຄວາມຕ້ອງການຂອງຜູ້ໃຊ້ລະບົບບໍ່ເຂົ້າກັບຈຸດປະສົງແລະຫາກ ມີການພັດທະນາລະບົບຕໍ່ໄປຈົນແລ້ວ ລະບົບງານທີ່ໄດ້ກໍ່ຈະບໍ່ກົງກັບຄາມຕ້ອງການຂອງຜູ້ໃຊ້ລະບົບຢ່າງແທ້ຈິງເຮັດໃຫ້ຕ້ອງມີຄວາມປັບປຸງຫຼືປ່ຽນແປງພາຍຫຼັງ.

ນັກວິເຄາະລະບົບສາມາດລວບລວມຄວາມຕ້ອງການຕ່າງໆໄດ້ຈາກການສັງເກດ, ການເຮັດວຽກຂອງຜູ້ໃຊ້, ການໃຊ້ເຕັກນິກ, ການສຳພາດ ຫຼື່ ການໃຊ້ແບບສອບຖາມ ການອ່ານເອກະສານກ່ຽວກັບການປະຕິບັດງານຂອງລະບົບງານປັດຈຸບັນລະບຽບກົດເກນຂອງບໍລີສັດແລະການມອບໝາຍຕໍາແໜ່ງ

ໜ້າທີ່ຮັບຜິດຊອບເຊິ່ງໃນຊ່ວງຂອງການເກັບກຳຂໍ້ມູນຄວາມຕ້ອງການ ກໍ່ຈະພົບກັບຜູ້ໃຊ້ໃນລະດັບຕ່າງໆທີ່ເຮັດໃຫ້ຮູ້ເຖິ່ງບັນຫາ ແລະ ແນວທາງການແກ້ໄຂບັນຫາທີ່ແນະນຳໂດຍຜູ້ໃຊ້ດັ່ງນັ້ນການເກັບກຳຄວາມຕ້ອງການຈຶ່ງເປັນກິດຈະກຳທີ່ສຳຄັນເພື່ອຄົ້ນຫາຄວາມຈິງ ແລະ ຕ້ອງສ້າງຄວາມຕ້ອງການເຊິ່ງກັນ ແລະ ກັນເພື່ອສະຫຼຸບອອກມາເປັນຂໍ້ກຳນົດທີ່ມີຄວາມຊັດເຈນ ໂດຍຂໍ້ກຳນົດເຫຼົ່ານີ້ເມື່ອຜູ້ໃຊ້ທີ່ກ່ຽວຂ້ອງໄດ້ອ່ານແລ້ວຈະຕ້ອງສຶກສາຄວາມໝາຍໄດ້ກົງກັນ.

ຫຼັງຈາກໄດ້ນໍາຄວາມຕ້ອງການຕ່າງໆ ມາສະຫຼຸບເປັນຂໍ້ກໍານົດທີ່ຊັດເຈນແລ້ວຂັ້ນຕອນຕໍ່ໄປກໍ່ຄືນໍາຂໍ້ກໍານົດເຫຼົ່ານັ້ນໄປພັດທະນາອອກມາເປັນຄວາມຕ້ອງການຂອງລະບົບໃໝ່ໂດຍເຕັກນິກທີ່ໃຊ້ກໍ່ຄືການພັດທະນາແບບຈໍາລອງຂະບວນການ (Process Model) ເຊິ່ງເປັນແຜນພາບທີ່ໃຊ້ອະທິບາຍເຖິງຂະບວນການທີ່ຕ້ອງດຳເນີນໃນລະບົບວ່າມີຫຍັງແດ່ ແລະ ຕໍ່ໄປກໍ່ດໍາເນີນການພັດທະນາແບບຈຳລອງຂໍ້ມູນ (Data Model) ເພື່ອອະທິບາຍເຖິງຂໍ້ມູນທີ່ຈັດເກັບໄວ້ສໍາລັບສະໜັບສະໜູນການເຮັກວຽກຕ່າງໆ.

ສະຫຼຸບໄລຍະຂອງການວິເຄາະລະບົບປະກອບມີກິດຈະກໍາຕ່າງໆດັ່ງນີ້:

- ວິເຄາະລະບົບງານປັດຈຸບັນ.

- ເກັບກໍາຄວາມຕ້ອງການໃນດ້ານຕ່າງໆ ແລະ ນໍາມາວິເຄາະເພື່ອສະຫຼຸບເປັນຂໍ້ກໍານົດຊັດເຈນ

- ນຳຂໍ້ກໍານົດມາພັດທະນາອອກມາເປັນຄວາມຕ້ອງການຂອງລະບົບໃໝ່.

- ສ້າງແບບຈຳລອງຂະບວນການຂອງລະບົບໃໝ່ໂດຍການແຕ້ມແຜນພາບກະແສຂໍ້ມູນ (Data Flow Diagram :DFD).

- ສ້າງແບບຈໍາລອງຂໍ້ມູນໂດຍການແຕ້ມ Entity Relationship Diagram: ERD.

* + 1. **ໄລ່ຍະການ 3 ການອອກແບບ (Design Phase)**

ໄລຍະການອອກແບບເປັນການພິຈາລະນາວ່າ ລະບົບລະດໍາເນີນການໄປໄດ້ແນວໃດເຊິ່ງກ່ຽວ ຂ້ອງກັບຍຸດທະວິທີການຂອງການອອກແບບທີ່ວ່າດ້ວຍການຕັດສິນໃຈວ່າຈະພັດທະນາລະບົບໃໝ່ດ້ວຍແນວທາງໃດເຊັ່ນພັດທະນາຂຶ້ນເອງ,ຊື້ໂປຣແກຣມສຳເລັດຮູບຫຼືວ່າຈ້າງບໍລິສັດພັດທະນາລະບົບໃຫ້ເປັນ

ຕົ້ນ ນອກຈາກນີ້ໄລ່ຍະການອອກແບບຈະກ່ຽວຂ້ອງກັບການອອກແບບ ທາງດ້ານສະຖາປັດຕະຍະກຳລະບົບທີ່ກ່ຽວຂ້ອງກັບອຸປະກອນຮາດແວ, ຊອບແວ ແລະ ເຄືອຂ່າຍ.

ການອອກແບບລາຍງານ (Out Design)(System Flowchart), ການອອກແບບໜ້າຈໍເພື່ອປະຕິສໍາພັນກັບຜູ້ໃຊ້ (User Interface), ການອອກແບບຜັງງານລະບົບ ເຊິ່ງລວມເຖິ່ງລາຍລະອຽດຂອງໂປຣແກມ (Specific Program), ຖານຂໍ້ມູນ (Database) ແລະ ຂໍ້ມູນທີ່ກ່ຽວຂ້ອງແນວໃດກໍ່ຕາມເຖິ່ງວ່າກິດຈະກຳບາງສ່ວນຂອງໄລຍະອອກແບບນີ້ສ່ວນໃຫ່ຍຈະຖືກດຳເນີນການໄປບາງສ່ວນແລ້ວ ໃນໄລຍະຂອງການວິເຄາະແຕ່ໄລຍະການອອກແບບນີ້ ຈະເນັ້ນເຖິ່ງການດຳເນີນການແກ້ໄຂບັນຫາແນວໃດຫຼາຍກ່ວາໂດຍການນຳຜົນຂອງແບບຈຳລອງທາງ Logical Model. ທີ່ໄດ້ຈາກການວິເຄາະມາພັດທະນາມາເປັນແບບຈໍາລອງທາງ Physical Model.

- ການວິເຄາະຈະເນັ້ນແກ້ໄຂບັນຫາຫຍັງແດ່.

- ການອອກແບບຈະເນັ້ນການແກ້ໄຂບັນຫາແນວໃດ.

- ສະຫຼຸບໄລຍະການອອກແບບ.

- ພິຈາລະນາແນວທາງໃນການພັດທະນາລະບົບ.

- ອອກແບບສະຖາປັດຕະຍະກໍາລະບົບ (Architecture Design).

- ອອກແບບຖານຂໍ້ມູນ (Database Design).

- ອອກແບບການສະແດງຜົນ (Output Design).

- ອອກແບບການປ້ອນຂໍ້ມູນ (Input Design).

- ອອກແບບສ່ວນຕິດຕໍໍກັບຜູ້ໃຊ້ (User Interface).

- ສ້າງຕົ້ນແບບ (Prototype).

- ອອກແບບໂປຣແກມ (Structure Chart).

* + 1. **ໄລ່ຍະການທີ່ 4 ການນຳໄປໃຊ້ (Implementation Phase)**

ໃນໄລຍະການນນໍາໄປໃຊ້ຈະເຮັດໃຫ້ລະບົບເກີດຜົນຂຶ້ນມາໂດຍການສ້າງລະບົບທົດສອບລະ

ບົບ ແລະ ການຕິດຕັ້ງລະບົບໂດຍຈຸດປະສົງຫຼັກຂອງກິດຈະກຳໃນໄລຍະນີ້ ບໍ່ແມ່ນພຽງຄວາມໜ້າເຊື່ອຖືຂອງລະບົບ ຫຼື ລະບົບສາມາດເຮັດວຽກໄດ້ດີເທົ່ານັ້ນ ແຕ່ຕ້ອງໝັ້ນໃຈວ່າຜູ້ໃຊ້ລະບົບຕ້ອງໄດ້ຮັບການເຝິກອົບຮົມເພື່ອໃຊ້ງານລະບົບ ແລະ ຄວາມຄາດຫວັງໃນອົງກອນທີ່ຕ້ອງການຜົນຕອບແທນໃນດ້ານດີກັບການໃຊ້ລະບົບໃໝ່ລຳດັບກິດຈະກຳຕ່າງໆທຸກກິດຈະກຳຕ້ອງເຂົ້າມາດຳເນີນການຮ່ວມກັນໃນລະໄຍນີ້ເພື່ອໃຫ້ລະບົບການປະຕິບັດງານໄດ້ຮັບຄວາມປະສົບຜົນສຳເລັດໄດ້ໂດຍດີ.

ສະຫຼຸບໄລຍະການນໍາໄປໃຊ້ຈະປະກອບມີກິດຈະກໍາຕ່າງໆດັ່ງຕໍ່ໄປນີ້:

- ສ້າງລະບົບຂຶ້ນມາດ້ວຍການຂຽນໂປຣແກມ.

- ກວດສອບຄວາມຖືກຕ້ອງທາງດ້ານ Verification ແລະ Validation ແລະດໍາເນີນການທົດສອບລະບົບ.

- ແປງຂໍ້ມູນ (Convert Data).

- ຕິດຕັ້ງລະບົບ (System Installation) ແລະສ້າງເອກກະສານຄູ່ມື.

- ຝຶກອົບຮົມຜູ້ໃຊ້ ແລະ ປະເມີນຜົນລະບົບໃໝ່.

ສໍາລັບການສ້າງລະບົບຫຼື່ການຂຽນໂປຣແກມນັ້ນສາມາດໃຊ້ວິທີການຂຽນໂປຣແກມດ້ວຍພາສາຄອມພິວເຕີເຊັ່ນ: ການໃຊ້ພາສາ Nodejs, Visual Basic, C#, PHP, Java... ນອກຈາກນີ້ຫຍັງມີເຕັກນິກອື່ນໆເຊັ່ນ: ເຄື່ອງມືໃນການພັດທະນາ Application ເຊິ່ງເປັນຊອບແວທີ່ເປັນແຫຼ່ງລວມຂອງເຄື່ອງມືຕ່າງໆທີ່ໃຊ້ເພື່ອພັດທະນາ Applicationເຮັດໃຫ້ຜູ້ຂຽນໂປຣແກມບໍ່ເຮັດວຽກໜັກຄືເມື່ອກ່ອນມີແຕ່ຮຽນຮູ້ ແລະປະຍຸກໃຊ້ເຄື່ອງມືເຫຼົ່ານັ້ນກໍ່ສາມາດພັດທະນາລະບົບງ່າຍຂຶ້ນ.

* + 1. **ໄລຍະທີ 5 ການບໍາລຸງຮັກສາ (Maintenance Phase)**

ໂດຍປົກກະຕິແລ້ວໄລຍະການບໍາລຸງຮັກສາຈະບໍ່ນໍາເຂົ້າໄປລວມໃນສ່ວນຂອງ SDLC ຈົນກະ ທັ້ງລະບົບມີການຕິດຕັ້ງເພື່ອໃຊ້ງານແລ້ວເທົ່ານັ້ນໄລຍະນີ້ຈະໃຊ້ເວລາຍາວນານທີ່ສຸດເມື່ອທຽບກັບໄລ ຍະອື່ນໆ ທີ່ຜ່ານມາເນື່ອງຈາກລະບົບຈະຕ້ອງໄດ້ຮັບການບຳລຸງຮັກສາຕະຫຼອດໄລຍະເວລາທີ່ມີການໃຊ້ລະບົບ ສິ່ງທີ່ຄາດຫວັງຂອງໜ່ວຍງານກໍ່ຄືຕ້ອງການໃຫ້ລະບົບໃຊ້ງານຍາວນານຫຼາຍປີ ລະບົບສາມາດຮອງຮັບເຕັກໂນໂລຊີໃໝ່ໆໃນອະນາຄົດໄດ້ ດັ່ງນັ້ນໃນຊ່ວງໄລຍະເວລາດັ່ງກາວຈຶ່ງສາມາດເພີ່ມເຕີ່ມຄວາມສາມາດຂອງລະບົບໃຫ້ມີປະສິດຕິພາບສູງຂຶ້ນ ພ້ອມທັງການແກ້ໄຂປັບປຸງໂປຣແກມໃນກໍລະນີທີ່ເຫັນຂໍ້ຜິດພາດ.

- ການບໍາລຸງຮັກສາລະບົບ (System Maintenannce).

- ການເພີ່ມເຕີມຄວາມສາມາດໃໝ່ໆເຂົ້າໃນລະບົບ (Enhance System).

- ສະໜັບສະໜູນງານຂອງຜູ້ໃຊ້ (Support the User).

ຈາກໄລຍະຕ່າງໆຕາມຂັ້ນຕອນການພັດທະນາລະບົບຕາມແບບແຜນຂອງ SDLC ຈະເຫັນວ່າມີການໃຊ້ຄໍາວ່າໄລຍະ ແລະກິດຈະກໍາເຊິ່ງສາມາດອະທິບາຍລາຍລະອຽດເພື່ອໃຫ້ເກີດຄວາມເຂົ້າໃຈກົງກັນດັ່ງນີ້:

- ໄລຍະ(Phase) ຄືກຸ່ມກິດຈະກໍາທີ່ກ່ຽວຂ້ອງກັນ.

- ກິດຈະກໍາ (Activity) ຄືກຸ່ມຂອງງານທີ່ກຽວຂ້ອງກັນ.

- ງານ (Task) ຄືງານທີ່ດໍາເນີນການເຊິ່ງຖືເປັນງານທີ່ນ້ອຍທີ່ສຸດ.

* 1. **ເຄື່ອງມືທີ່ໃຊ້ໃນການພັດທະນາ**
     1. **Hardware**
* ຄອມພິວເຕີ Asus Intel Core(TM) I3-7020U @2.3GHz, RAM 8GB 2133 MHz
* USB 16GB 1 ອັນ
  + 1. **Software**
* ລະບົບປະຕິບັດການ Windows 10 Professional 64 Bit
* Microsoft Visio 2016 ໃຊ້ແຕ້ມແຜນວາດການໄຫຼຂໍ້ມູນ (DFD, ER, Flowchart)
* Robo 3T ໃຊ້ຈັດການຖານຂໍ້ມູນ
* Visual Studio Code ໃຊ້ຂຽນໂຄດດ້ວຍພາສາ JavaScript
* MS Office 2013 Professional ໃຊ້ເພື່ອສ້າງເອກະ ຕ່າງໆ, ນຳສະເໜີ ແລະ ເຮັດປຶ້ມບົດຈົບຊັ້ນ
  1. **ການວິເຄາະຂໍ້ມູນ**

### ແຜນວາດລວມຂອງລະບົບ (System Overview)

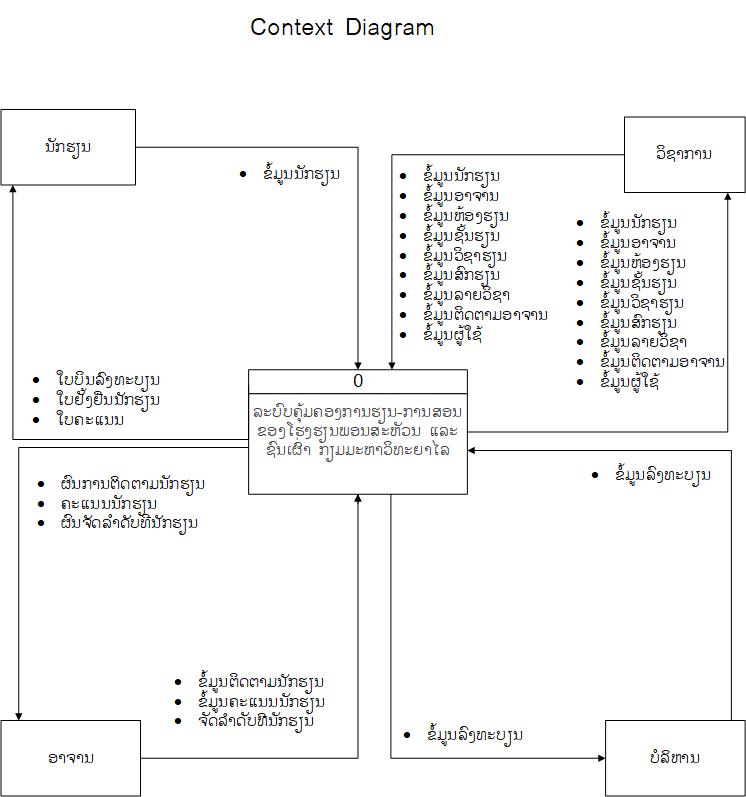
ແຜນວາດທີ່ x ແຜນວາດລວມຂອງລະບົບ (System Overview)

* + 1. **ຕາຕະລາງສະແດງລາຍລະອຽດຕ່າງໆກ່ຽວກັບລະບົບ**

|  |  |  |
| --- | --- | --- |
| External Entity | Process | Data Store |
| ອາຈານ  ວິຊາການ  ບໍລິຫານ  ຜູ້ອຳນວຍການ  ນັກຮຽນ | 1. **ຈັດການຂໍ້ມູນພື້ນຖານ**    1. ຈັດການຂໍ້ມູນອາຈານ    2. ຈັດການຂໍ້ມູນນັກຮຽນ    3. ຈັດການຂໍ້ມູນລາຍວິຊາ    4. ຈັດການຂໍ້ມູນວິຊາສອນ    5. ຈັດການຂໍ້ມູນສົກຮຽນ    6. ຈັດການຂໍ້ມູນຜູ້ໃຊ້    7. ຈັດການຂໍ້ມູນຫ້ອງ    8. ຈັດການຂໍ້ມູນຊັ້ນ    9. ຈັດການຂໍ້ມູນຄ່າທຳນຽມ 2. **ລົງທະບຽນ**    1. ກວດສອບ    2. ລົງທະບຽນຮຽນ 3. **ຈັດການຄະແນນ**    1. ບັນທຶກຄະແນນ    2. ຄົ້ນຫາຄະແນນ 4. **ປະເມີນຜົນ**    1. ຈັດລຳດັບທີ    2. ຕິດຕາມປະເມີນນັກຮຽນ    3. ເລື່ອນຊັ້ນ 5. **ລາຍງານ**    1. ຊີວະປະຫວັດນັກຮຽນ    2. ສະຖິຕິນັກຮຽນ    3. ໃບຢັ້ງຢືນການຮຽນ (ລາວ-ອັງກິດ)    4. ໃບຄະແນນ (ລາວ-ອັງກິດ)    5. ວັນຂາດນັກຮຽນ | D1 ຂໍ້ມູນອາຈານ  D2 ຂໍ້ມູນນັກຮຽນ  D3 ຂໍ້ມູນຊັ້ນຮຽນ  D4 ຂໍ້ມູນຫ້ອງຮຽນ  D5 ຂໍ້ມູນລົງທະບຽນ  D6 ຂໍ້ມູນຄະແນນ  D7 ຂໍ້ມູນລາຍວິຊາ  D8 ຂໍ້ມູນວິຊາສອນ  D9 ຂໍ້ມູນຕິດຕາມນັກຮຽນ  D10 ຂໍ້ມູນປະເມີນນັກຮຽນ  D11 ຂໍ້ມູນສົກຮຽນ  D12 ຂໍ້ມູນຜູ້ໃຊ້  D13 ຂໍ້ມູນຄ່າທຳນຽນຮຽນ  D14 ຂໍ້ມູນບ້ານ  D15 ຂໍ້ມູນເມືອງ  D16 ຂໍ້ມູນແຂວງ  D17 ຂໍ້ມູນຊົນເຜົາ |

ຕາຕະລາງທີ່ x ຕາຕະລາງສະແດງລາຍລະອຽດຕ່າງໆກ່ຽວກັບລະບົບ

* + 1. **ແຜນວາດເນື້ອຫາ (Context Diagram)**



ແຜນວາດທີ່ x: ແຜນວາດເນື້ອຫາ

* + 1. **ແຜນພາບ Process Hierarchy Chart (PHC)**

1. ຈັດການຂໍ້ມູນພື້ນຖານ
   1. ຈັດການຂໍ້ມູນອາຈານ
   2. ຈັດການຂໍ້ມູນນັກຮຽນ
   3. ຈັດການຂໍ້ມູນລາຍວິຊາ
   4. ຈັດການຂໍ້ມູນວິຊາສອນ
   5. ຈັດການຂໍ້ມູນສົກຮຽນ
   6. ຈັດການຂໍ້ມູນຜູ້ໃຊ້
   7. ຈັດການຂໍ້ມູນຫ້ອງ
   8. ຈັດການຂໍ້ມູນຊັ້ນ
   9. ຈັດການຂໍ້ມູນຄ່າທຳນຽມ
2. ລົງທະບຽນ
   1. ກວດສອບ
   2. ລົງທະບຽນຮຽນ
3. ຈັດການະແນນ
   1. ບັນທຶກຄະແນນ
   2. ຄົ້ນຫາຄະແນນ

ລະບົບຄຸ້ມຄອງການຮຽນ-ການສອນຂອງໂຮງຮຽນ

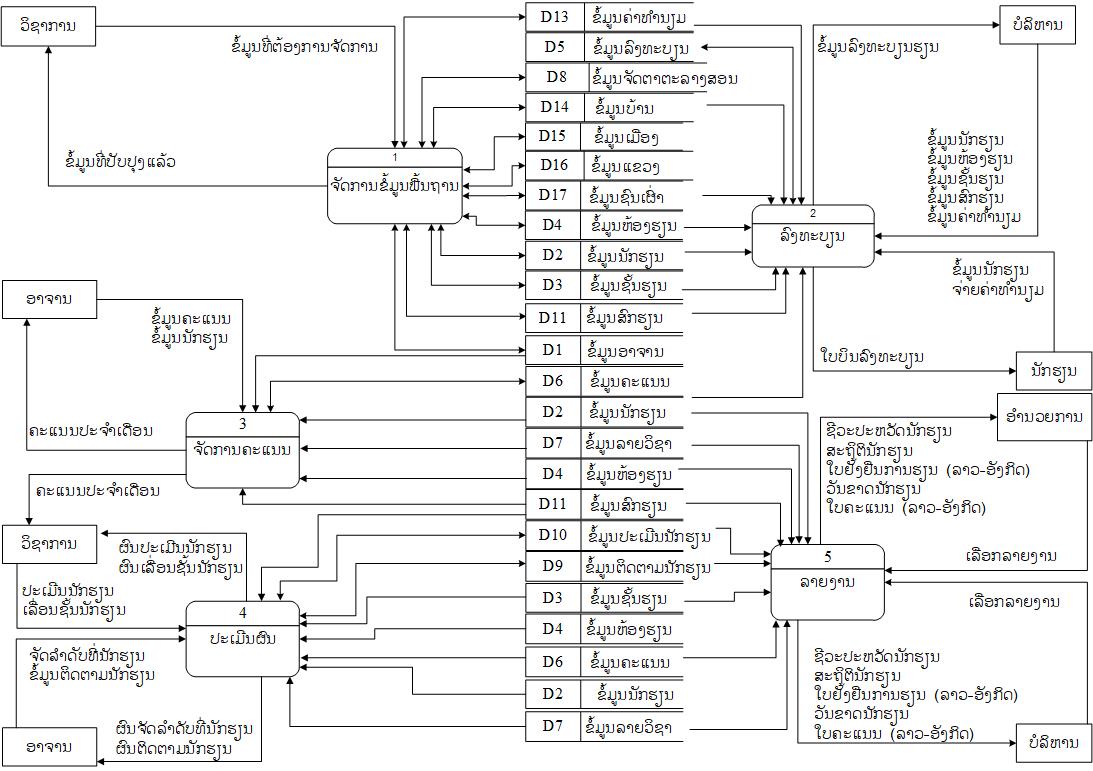
ພອນສະຫັວນ ແລະ ຊົນເຜົ່າກຽມມະຫາວິທະຍາໄລ

1. ປະເມິນຜົນ

3.3 ຈັດລຳດັບທີ

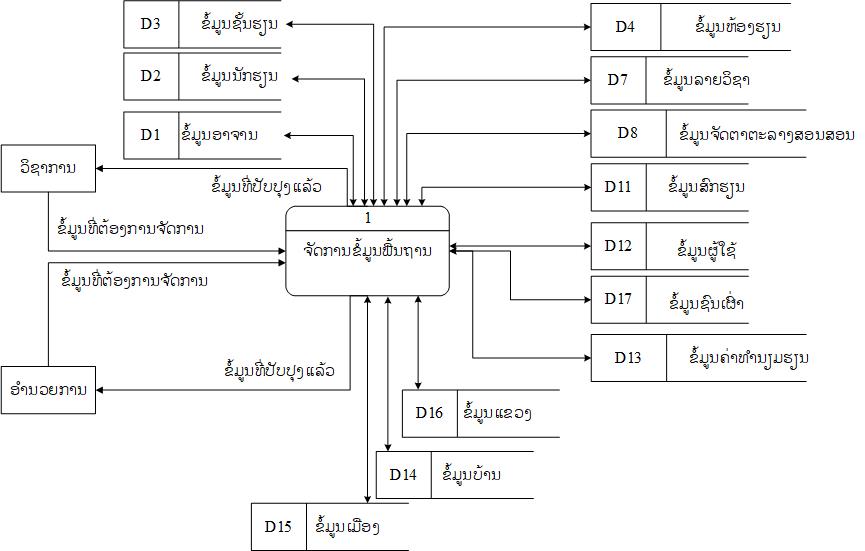
* 1. ຕິດຕາມປະເມີນນັກຮຽນ
  2. ເລືອນຊັ້ນ

1. ລາຍງານ
   1. ຊີວະປະຫວັດນັກຮຽນ
   2. ສະຖິຕິນັກຮຽນ
   3. ໃບຢັ້ງຢືນການຮຽນ(ລາວ-ອັງກິດ)
   4. ໃບຄະແນນ(ລາວ-ອັງກິດ)
   5. ຄະແນນນັກຮຽນ
   6. ວັນຂາດນັກຮຽນ
      1. **ແຜນວາດການໄຫຼຂໍ້ມູນ ( Data flow Diagram: DFD )**
2. ແຜນວາດລວມການໄຫຼຂໍ້ມູນລະດັບ 1 ຂອງແຕ່ລະ Process



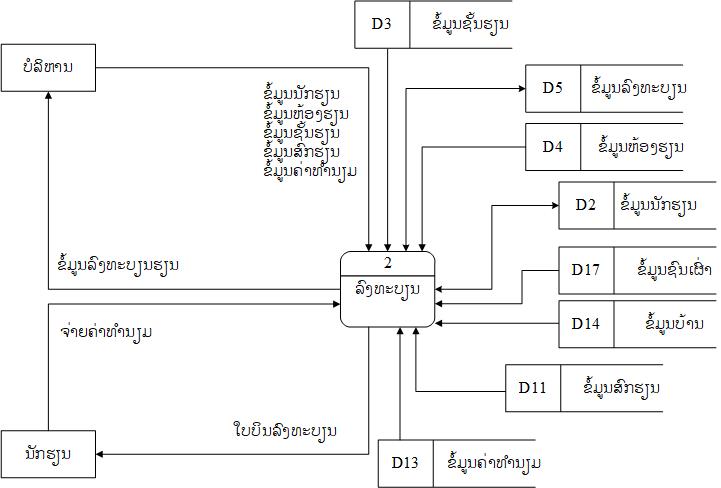
ແຜນວາດທີ່ x: ແຜນວາດການໄຫຼຂໍ້ມູນ

1. ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 ຂອງ Process ທີ່ 1



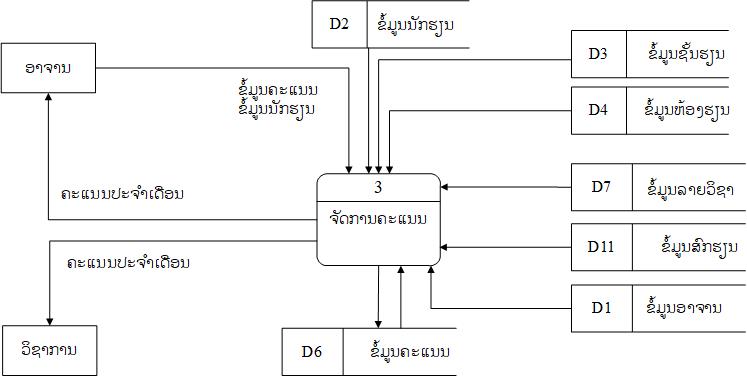
ແຜນວາດທີ່ x: ແຜນວາດການໄຫຼຂໍ້ມູນ

1. ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 ຂອງ Process ທີ່ 2

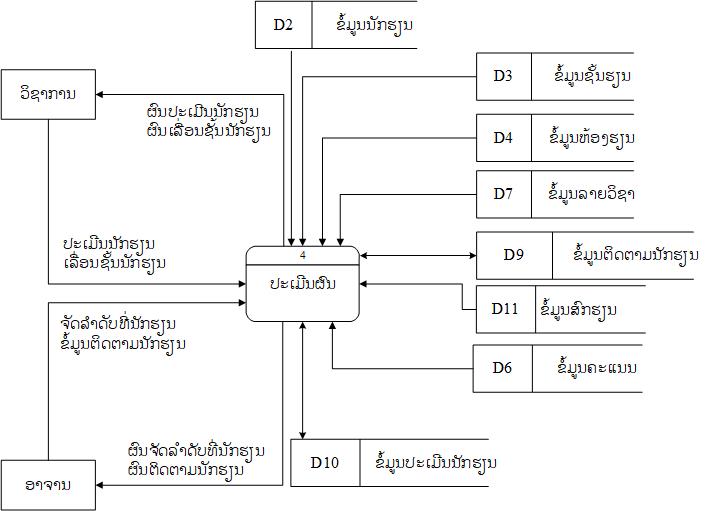


ແຜນວາດທີ່ x: ແຜນວາດການໄຫຼຂໍ້ມູນ

1. ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 ຂອງ Process ທີ່ 3

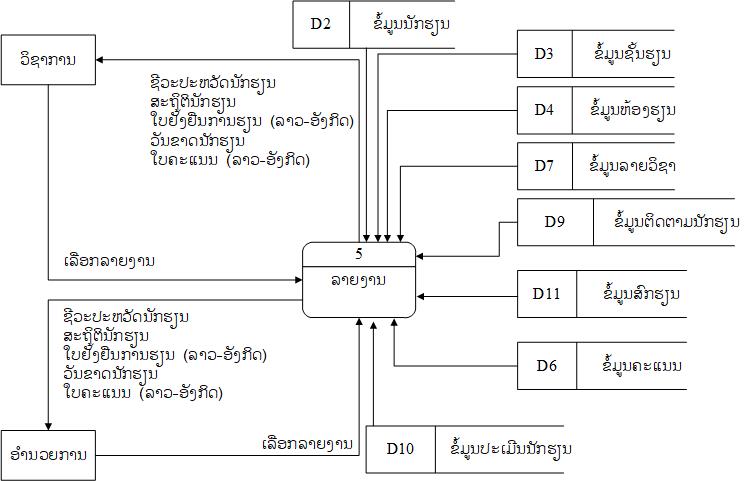


ແຜນວາດທີ່ x: ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ

1. ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 ຂອງ Process ທີ່ 4

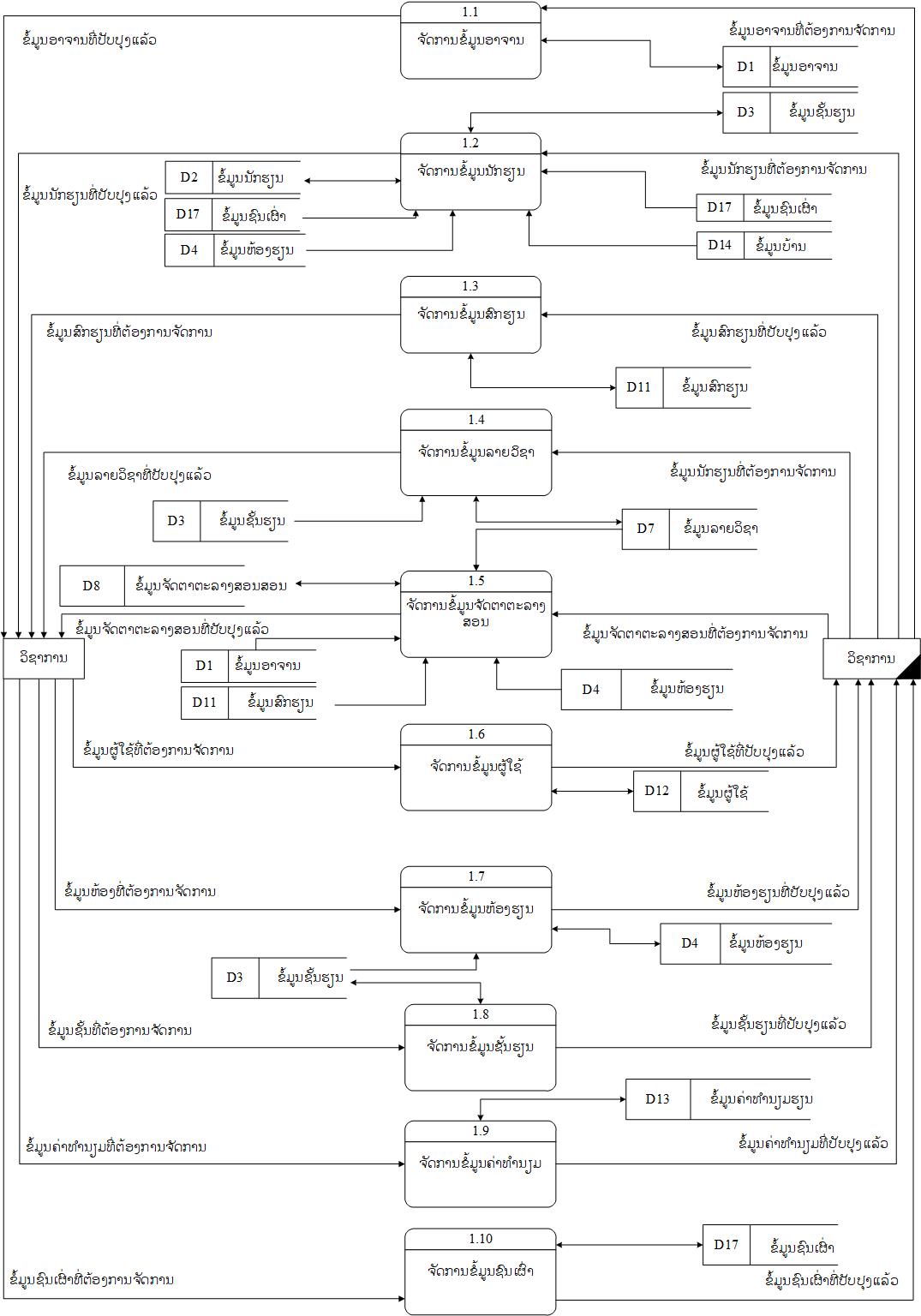
ແຜນວາດທີ່ 8: ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 Process ທີ່ 3

1. ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 1 Process ທີ່ 5



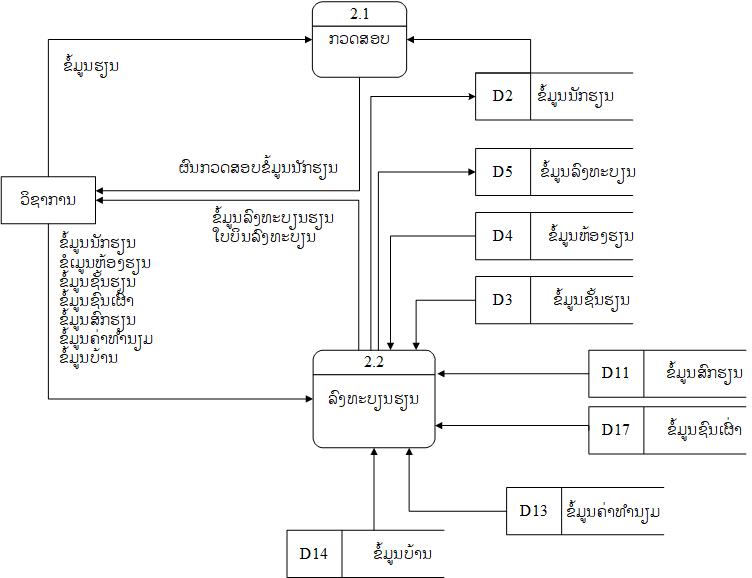
ແຜນວາດທີ່ 9: ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ

1. ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 2 Process ທີ່ 1



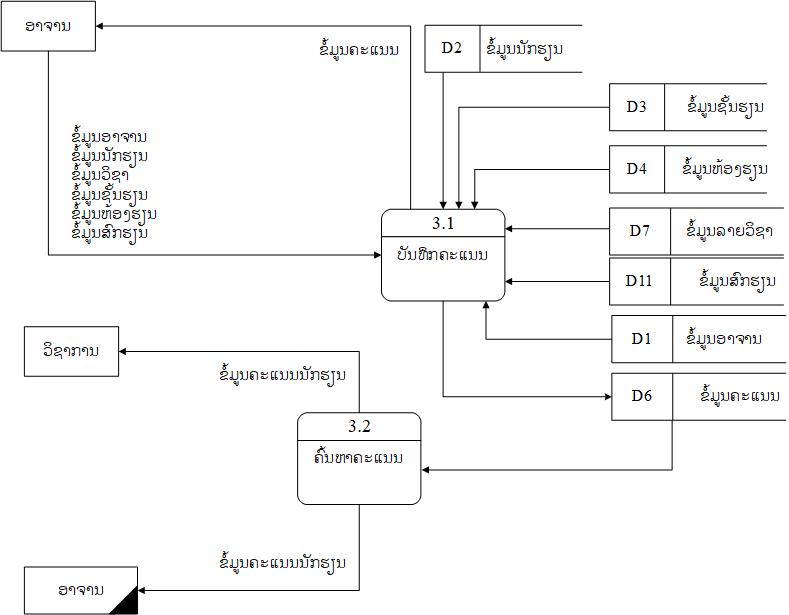
ແຜນວາດທີ່ 9: ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ

1. ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 2 Process ທີ່ 2



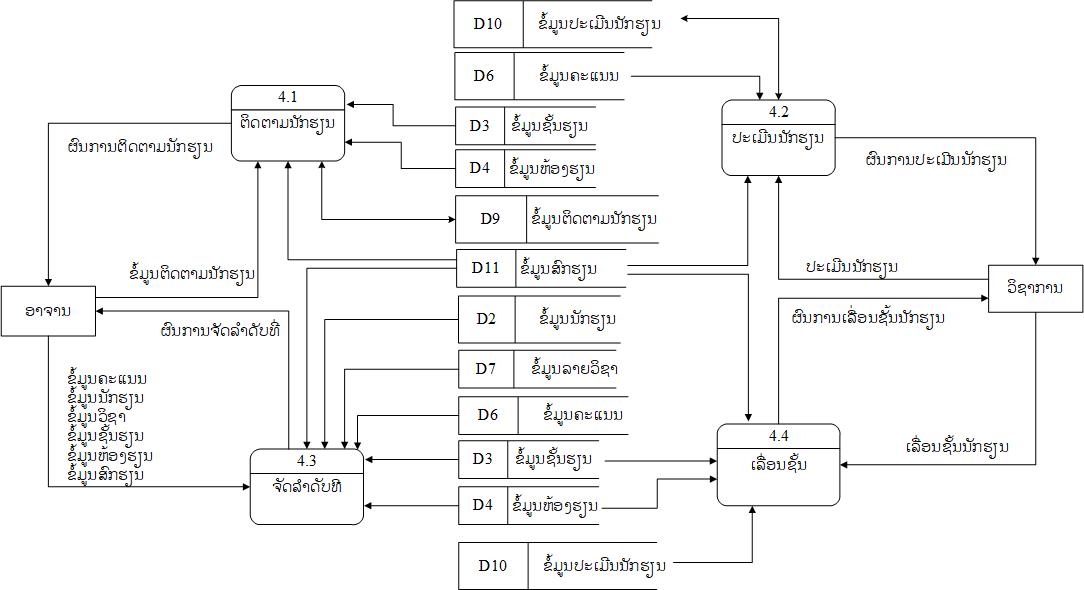
ແຜນວາດທີ່ x: ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ

1. ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 2 Process ທີ່ 3

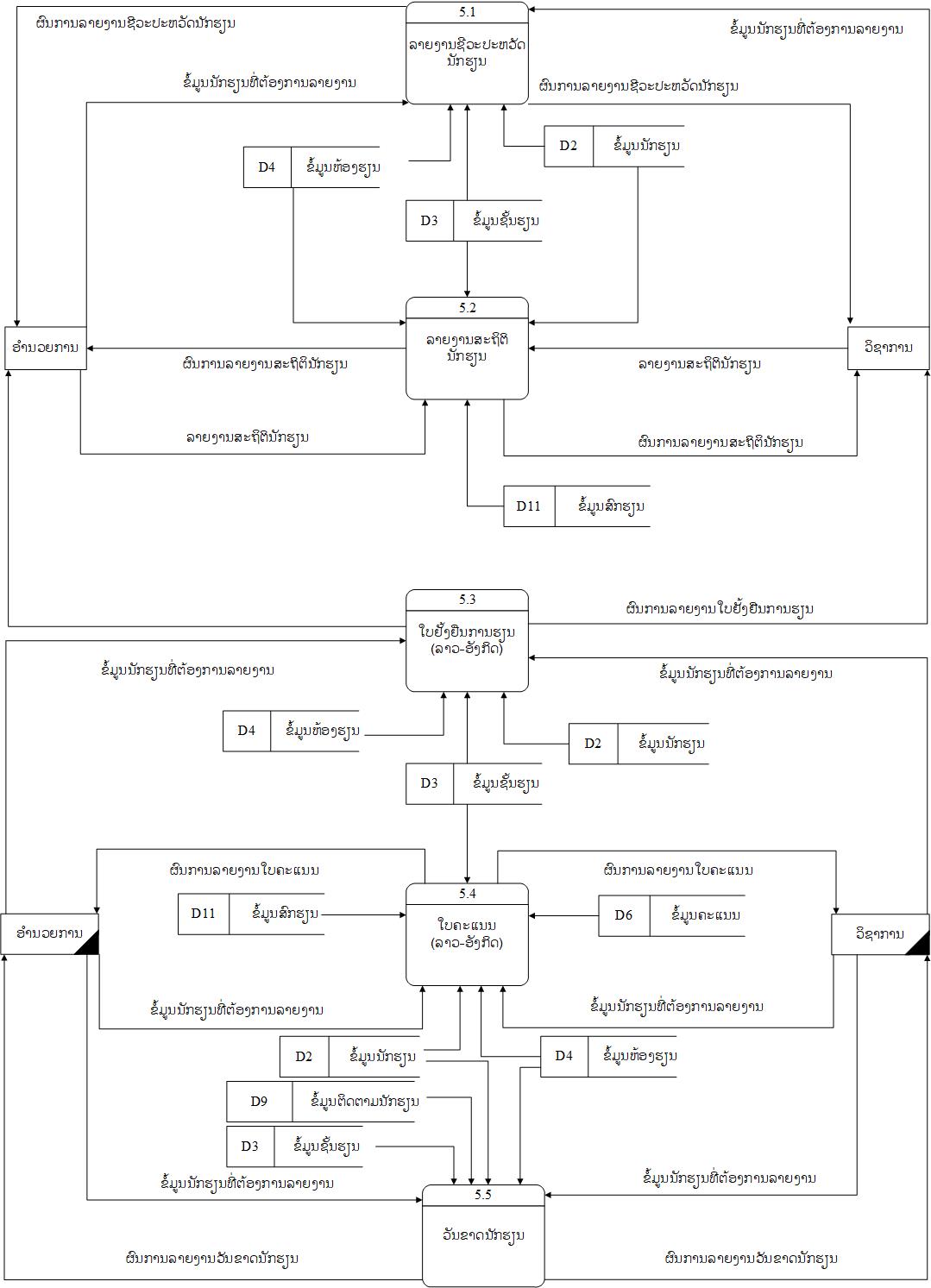


ແຜນວາດທີ່ x: ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ

1. ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 2 Process ທີ່ 4

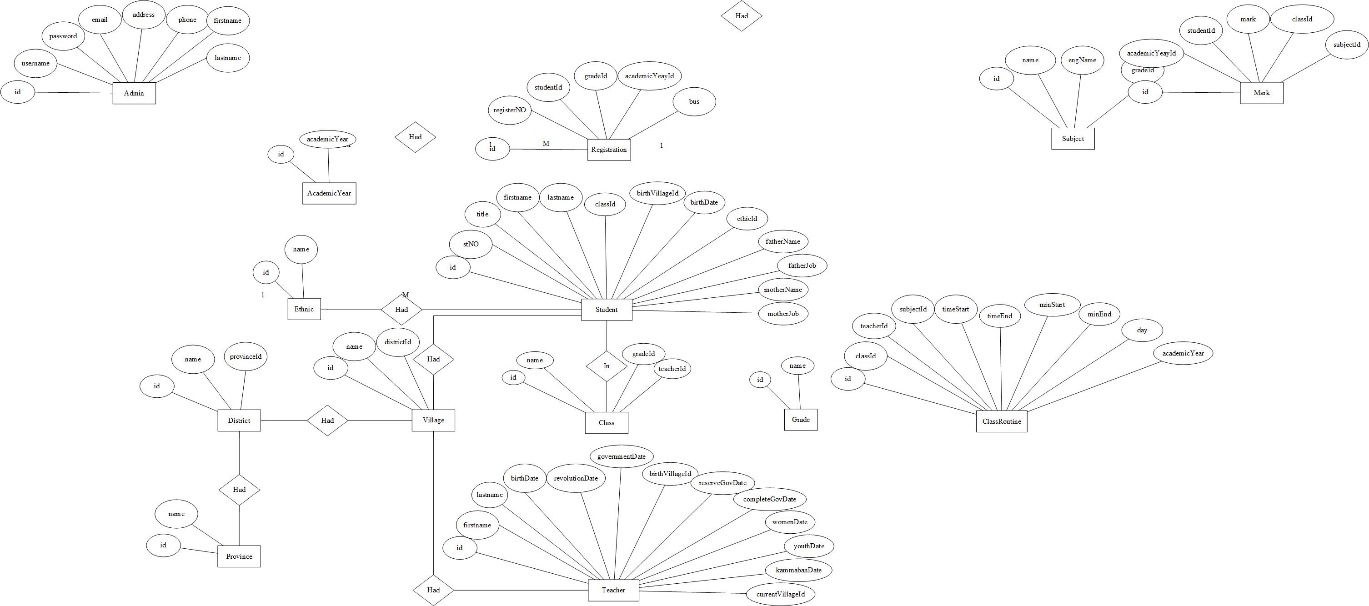


ແຜນວາດທີ່ x: ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ

1. ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ 2 Process ທີ່ 5

ແຜນວາດທີ່ x: ແຜນວາດການໄຫຼຂໍ້ມູນລະດັບ

* + 1. **ແຜນວາດຄວາມສຳພັນຂອງຂໍ້ມູນ (ER Diagram)**

****

### ວັດຈະນານຸກົມ (Data Dictionary )

1. ຕາຕະລາງຂໍ້ມູນບ້ານ (Village)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table: Village** | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| name | String | No |  | ຊື່ພະນັກງານ |  |
| districtId | ObjectId | No | Foreign Key | ນາມສະກຸນ |  |

1. ຕາຕະລາງຂໍ້ມູນເມືອງ (District)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table: District** | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| name | String | No |  | ຊື່ພະນັກງານ |  |
| provinceId | ObjectId | No | Foreign Key | ນາມສະກຸນ |  |

1. ຕາຕະລາງຂໍ້ມູນແຂວງ (Province)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table: Province** | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| name | String | No |  | ຊື່ພະນັກງານ |  |

1. ຕາຕະລາງຂໍ້ມູນອາຈານ (Teacher)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** Teacher | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| fristname | String | No |  | ຊື່ພະນັກງານ |  |
| Lastname | String | No |  |  |  |
| email | String | No |  |  |  |
| tel | String | No |  |  |  |
| password | String | No |  |  |  |
| birthDate | Date | No |  |  |  |
| revolutionDate | Date | yes |  |  |  |
| governmentDate | Date | yes |  |  |  |
| Organization{} | Date | yes |  |  |  |
| reserveGovDate | Date | yes |  |  |  |
| completeGovDate | Date | yes |  |  |  |
| womenDate | Date | yes |  |  |  |
| youthDate | Date | yes |  |  |  |
| kammabanDate | Date | yes |  |  |  |
| imageUrl | string | yes |  |  |  |

1. ຕາຕະລາງຂໍ້ມູນນັກຮຽນ (Student)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** Student | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| stNO | String | No |  | ຊື່ພະນັກງານ |  |
| title | String | No |  |  |  |
| fristname | String | No |  |  |  |
| lastname | String | No |  |  |  |
| classld | ObjectId | No | Foreign Key |  |  |
| birthDate | Date | No |  |  |  |
| ethnicld | ObjectId | No | Foreign Key |  |  |
| Guardian{} | Guardian | No |  |  |  |
| fatherName | String | yes |  |  |  |
| fatherJob | String | yes |  |  |  |
| motherName | String | yes |  |  |  |
| MotherJob | String | yes |  |  |  |

1. ຕາຕະລາງຂໍ້ມູນຊົນເຜົາ (Ethnic)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** Ethnic | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| name | String | No |  | ຊື່ພະນັກງານ |  |

1. ຕາຕະລາງຂໍ້ມູນການລົງທະບຽນ (Registration)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** Registration | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| registerNO | Number | No |  | ຊື່ພະນັກງານ |  |
| stuentld | ObjectId | No | Foreign Key |  |  |
| gradeld | ObjectId | No | Foreign Key |  |  |
| academicyearld | ObjectId | No | Foreign Key |  |  |

1. ຕາຕະລາງຂໍ້ມູນຜູ້ໃຊ້ງານ (Admin)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** Admin | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| username | String | No |  | ຊື່ພະນັກງານ |  |
| email | String | No |  |  |  |
| password | String | No |  |  |  |
| firstName | String | No |  |  |  |
| lastName | String | No |  |  |  |

1. ຕາຕະລາງຂໍ້ມູນຄະແນນ (Mark)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** Mark | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| academicYearld | ObjectId | No | Foreign Key | ຊື່ພະນັກງານ |  |
| studentld | ObjectId | No | Foreign Key |  |  |
| mark | ObjectId | No |  |  |  |
| classld | ObjectId | No | Foreign Key |  |  |
| subjectld | ObjectId | No | Foreign Key |  |  |

1. ຕາຕະລາງຂໍ້ມູນຕິດຕາມນັກຮຽນ (TrackStudent)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** TrackStudent | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| absentHours | Number | No |  | ຊື່ພະນັກງານ |  |
| studentld | ObjectId | No | Foreign Key |  |  |
| subjectld | ObjectId | No | Foreign Key |  |  |
| academicld | ObjectId | No | Foreign KeyForeign Key |  |  |
| classld | ObjectId | No | Foreign Key |  |  |
| daet | Date | No |  |  |  |

1. ຕາຕະລາງຂໍ້ມູນຫ້ອງ (Class)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** Class | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| name | String | No |  | ຊື່ພະນັກງານ |  |
| gradeld | ObjectId | No | Foreign Key |  |  |
| teacherld | ObjectId | yes | Foreign Key |  |  |

1. ຕາຕະລາງຂໍ້ມູນຊັ້ນຮຽນ (Grade)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** Grade | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| name | String | No |  | ຊື່ພະນັກງານ |  |

1. ຕາຕະລາງຂໍ້ມູນສົກຮຽນ (AcademicYear)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** AcademicYear | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| academicYear | String | No |  | ຊື່ພະນັກງານ |  |

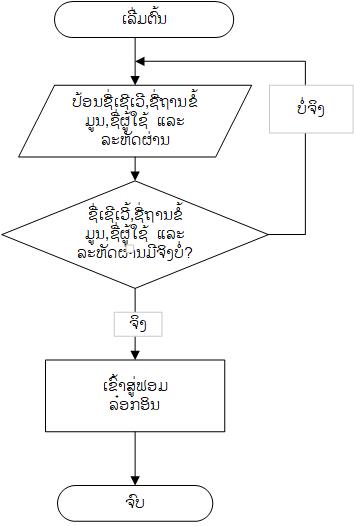
1. ຕາຕະລາງຂໍ້ມູນວິຊາ (Subject)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** Subject | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| name | String | No |  | ຊື່ພະນັກງານ |  |
| engName | String | yes |  |  |  |
| gradeld | ObjectId | No | Foreign Key |  |  |

1. ຕາຕະລາງຂໍ້ມູນຕາລາງຮຽນ (ClassRoutine)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Table:** ClassRoutine | | | | | |
| Field Name | Data Type | Allow Null | Key | Description | Reference |
| id | ObjectId | No | PK | ລະຫັດບ້ານ |  |
| teacher | String | No | Foreign Key | ຊື່ພະນັກງານ |  |
| classld | ObjectId | No | Foreign Key |  |  |
| sujectld | ObjectId | No | Foreign Key |  |  |
| time | String | No |  |  |  |
| day | String | No |  |  |  |
| work | Boolean | No |  |  |  |

# **ການອອກແບບໂປແກຣມ**

* 1. Flowchart ຂອງຟອມເຂົ້າສູ່ເຄື່ອງເຊີເວີ້



# **Flowchart ຂອງຟອມລ່ອກອິນເຂົ້າລະບົບ**

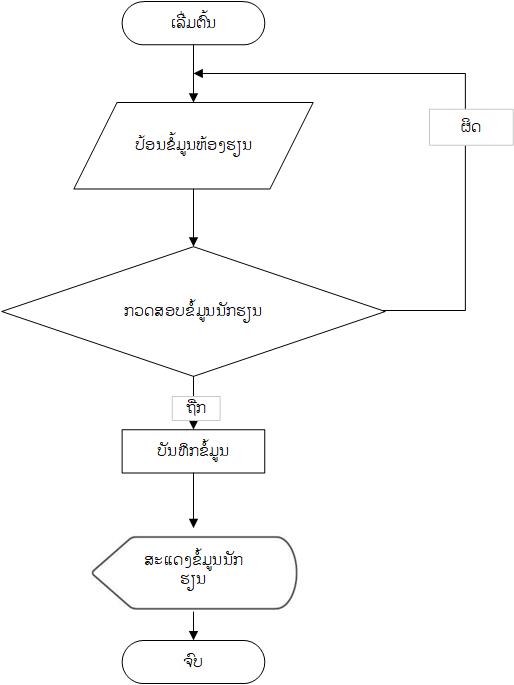
# **D:\PhonesavanhL-TMSDocs\Diagram\img\Drawing4.jpg**

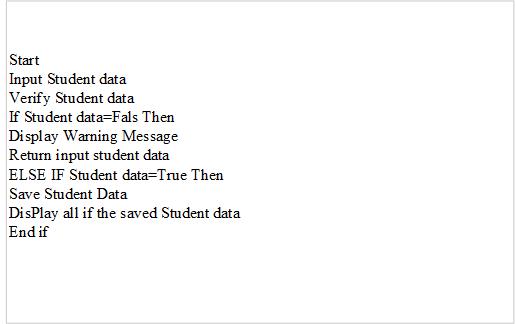
# **D:\PhonesavanhL-TMSDocs\Diagram\img\Drawing4.jpg**

# flowchart ຂອງຟອມລົງທະບຽນ

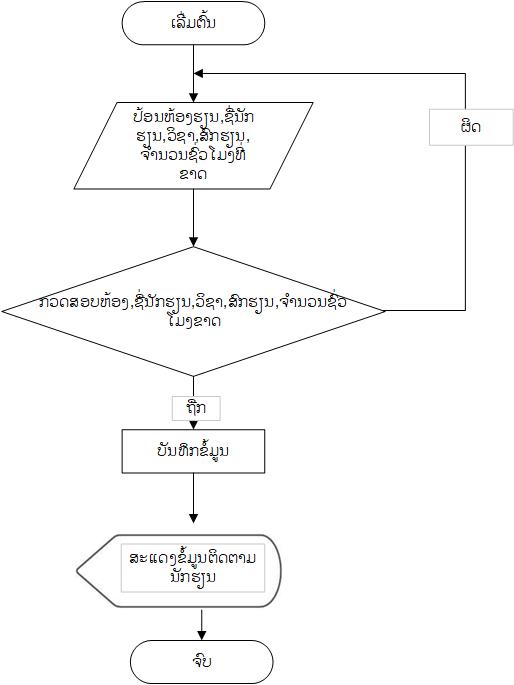
# **D:\PhonesavanhL-TMSDocs\Diagram\img\flowchart.jpg33.jpg**

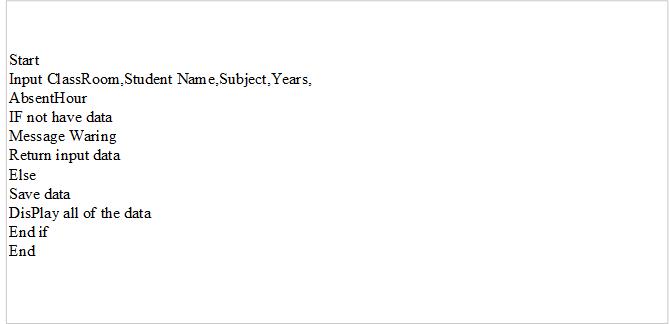
# **D:\PhonesavanhL-TMSDocs\Diagram\img\ໄຊແຕ້ມ.jpg 1.jpg**

* 1. Flowchart ຂອງຟອມຈັດການຂໍ້ມູນນັກຮຽນ

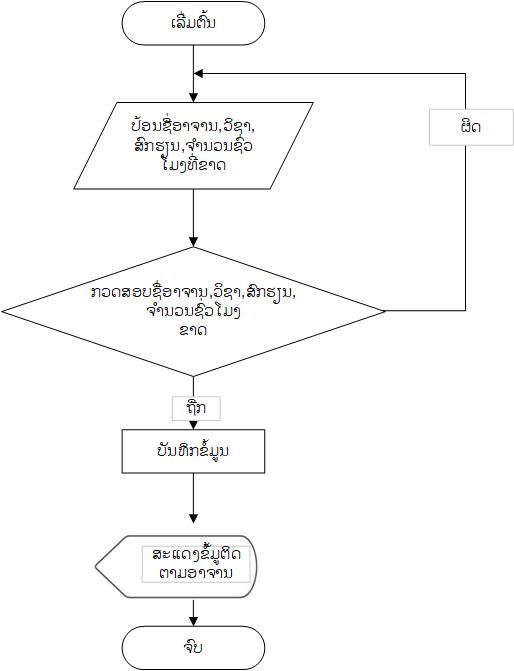


* 1. Flowchart ຂອງຟອມຕິດຕາມນັກຮຽນ



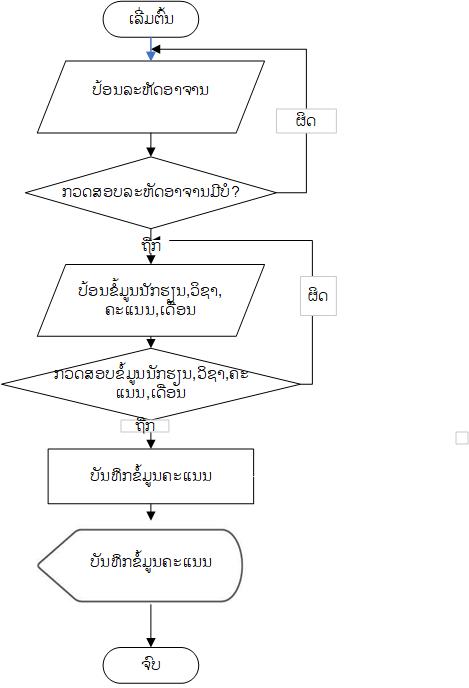


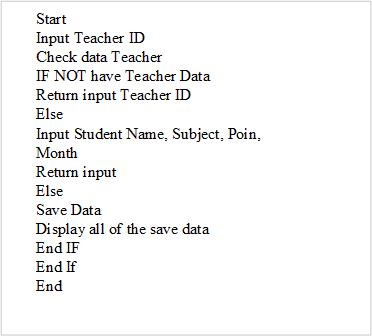
* 1. Flowchart ຂອງຟອມຕິດຕາມອາຈາານ



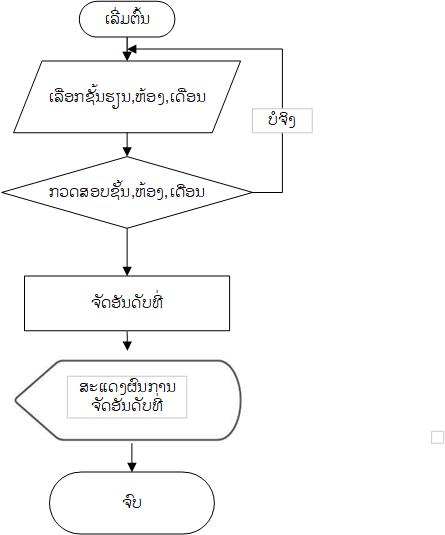


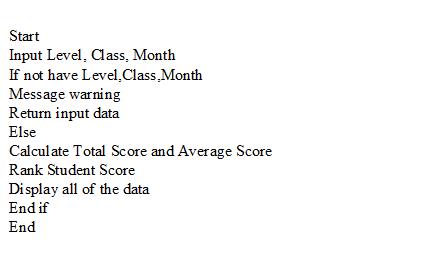
* 1. Flowchart ຂອງຟອມບັນທຶກຄະແນນ



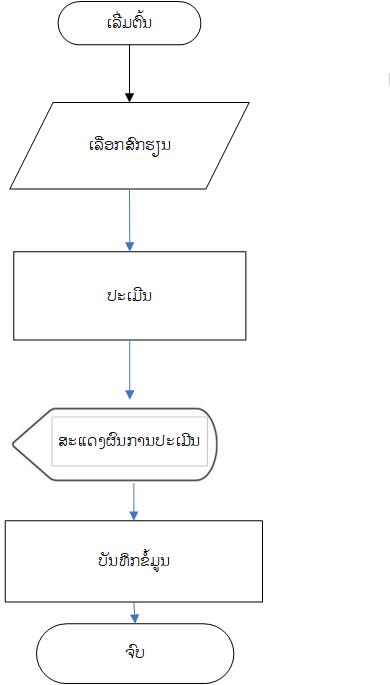


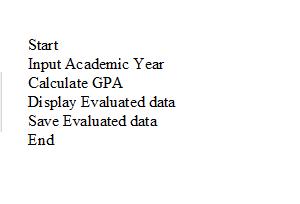
* 1. Flowchar ຂອງຟອມຈັດລຳດັບທີ່



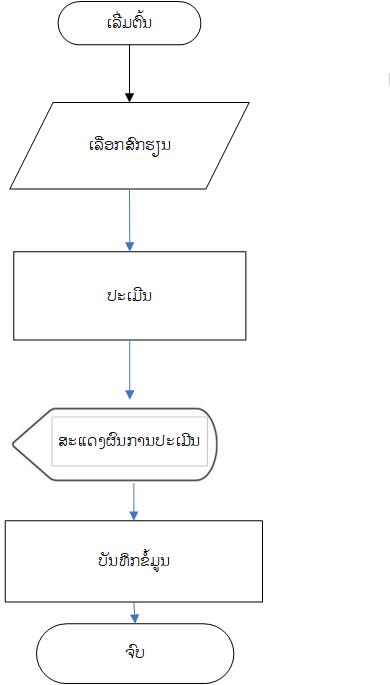


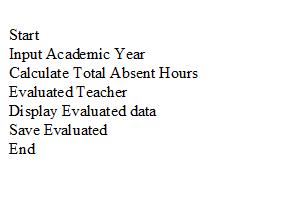
* 1. Flowchart ຂອງຟອມປະເມີນນັກຮຽນ





* 1. Flowchart ຂອງຟອມປະເມີນອາຈານ





# **ບົດທີ 4**

# **ຜົນຂອງການວິເຄາະຂໍ້ມູນ ແລະ ການອະທິບາຍຜົນ**

### 4.1 ໜ້າຕ່າງການ (Login Form)

ເມື່ອເປີດໂປຣແປຣມຂຶ້ນມາກໍ່ຈະເຫັນວ່າເຂົ້າສູ່ລະບົບເປັນໜ້າທຳອິດຈາກນັ້ນຈຶ່ງທຳການເຂົ້າສູ່ລະບົບດ້ວຍຊື່ ແລະ ລະຫັດເຂົ້າໃຊ້ເພື່ອເຂົ້າໄປດ້ານໃນຂອງໂປຣແກຣມດັ່ງຮູບລຸ່ມນີ້:

- ວິທີການເຂົ້າສູ່ລະບົບມີຄື :

1) ໃສ່ຊື່ຜູ້ໃຊ້ລະບົບ.

2) ໃສ່ລະຫັດຜ່ານ.

3) ກົດປຸ່ມເພື່ອເຂົ້າລະບົບ.

4) ຖ້າຊື່ກັບລະຫັດຖືກຈະເປີດນຳໃຊ້ໂປຣແກຣມໄດ້ປົກກະຕິແລ້ວ.

5) ຖ້າຊື່ກັບລະຫັດຜິດຈະມີຂໍ້ຄວາມຂື້ນມາວ່າ ກະລຸນາກວດສອບຂໍ້ມູນຂອງທ່ານໃຫ້ຖືກຕ້ອງ ແລ້ວກົດປຸ່ມ OK

6) ຖ້າບໍ່ເຂົ້າລະບົບແມ່ນກົດປຸ່ມອອກລະບົບ.

**ຮູບທີ 4.**1 ຟອມຫຼັກ (Login)

### 4.2 ຟ້ອນຫຼັກ (Main Form)

ຟອມຫຼັກແມ່ນຟອມທີ່ເຮົາສາມາດເຂົ້າຫາທຸກໆຟອມໄດ້ເຊັ່ນ ຟອມຈັດການຂໍ້ມູນ, ຟອມປ້ອນຂໍ້ມູນ, ຟອມການບໍລິການ, ຟອມລາຍຮັບ, ຟອມລາຍຈ່າຍ, ຟອມລາຍງານ.

ຮູບທີ 4.2 ຟອມຫຼັກ (Main Form)

### 4.3 ໜ້າຟອມຈັດການຫ້ອງພັກຂອງໂປຣມແກຣມ

ເມື່ອເຮົາເຂົ້າມາໃນໜ້າຫຼັກຂອງໂປຣແກຣມແລ້ວຖ້າເລືອກການຈັດການຫ້ອງພັກເຮົາກໍ່ຈະເຫັນດັ່ງຮູບລຸ່ມນີ້: ເຊັ່ນວ່າເຮົາປ້ອນຂໍ້ມູນຫ້ອງ, ລະຫັດຫ້ອງ, ຊື່ເບີຫ້ອງ, ສະຖານະ, ຂໍ້ມູນປະເພດຫ້ອງ.

ຮູບທີ 4.3 ຮູບໜ້າຟອມຫ້ອງພັກ

### 4.4 ໜ້າຟອມຈັດການປະເພດຫ້ອງຂອງໂປຣມແກຣມ

ເມື່ອເຮົາເຂົ້າມາໃນໜ້າຫຼັກຂອງໂປຣແກຣມແລ້ວຖ້າເລືອກການຈັດການປະເພດຫ້ອງພັກເຮົາກໍ່ຈະເຫັນດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.4 ຮູບໜ້າຟອມປະເພດຫ້ອງພັກ

### 4.5 ໜ້າຟອມຈັດການລູກຄ້າຂອງໂປຣມແກຣມ

ເມື່ອເຮົາເຂົ້າມາໃນໜ້າຫຼັກຂອງໂປຣແກຣມແລ້ວຖ້າເລືອກການຈັດການຂໍ້ມູນລູກຄ້າເຮົາກໍ່ສາມາດຄົ້ນຫາຕາມລະຫັດ ຫຼື ຄົ້ນຫາຕາມຊື້.

ຮູບທີ 4.5 ຮູບໜ້າຟອມຈັດການຂໍ້ມູນລູກຄ້າ

### 4.6 ໜ້າຟອມການຈັດການຂໍ້ມູນພະນັກງານ

ເມື່ອເຮົາເຂົ້າມາໃນໜ້າຫຼັກຂອງໂປຣແກຣມແລ້ວຖ້າເລືອກການຈັດການຂໍ້ມູນພະນັກງານເຮົາກໍ່ສາມາດປ້ອນຂໍ້ມູນພະນັກງານບໍ່ວ່າຈະຄົ້ນຫາດ້ວຍການໃສ່ລະຫັດ ຫຼື ຄົ້ນຫາຈາກຊື້, ການເພີ່ມ, ແກ້ໄຂ, ບັນທືກ ແລະ ລຶບກໍ່ໄດ້.

ຮູບທີ 4.6 ຮູບໜ້າຟອມຈັດການຂໍ້ມູນພະນັກງານ

### 4.7 ໜ້າຟອມການບໍລິການ

ເມື່ອເຮົາເຂົ້າມາໃນໜ້າຫຼັກຂອງໂປຣແກຣມແລ້ວຖ້າເລືອກການຈັດການບໍລິການເຮົາກໍ່ຈະເຫັນລາຍລະອຽດດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.7 ຮູບໜ້າຟອມການບໍລິການ

### 4.8 ໜ້າຟອມການຈັດການ ການຈອງ

ເມື່ອເຮົາເຂົ້າມາໃນໜ້າການຈອງຂອງໂປຣແກຣມແລ້ວຖ້າເລືອກການຈັດການຈອງເຮົາກໍ່ຈະເຫັນລາຍລະອຽດດັ່ງຮູບລຸ່ມນີ້

ຮູບທີ 4.8 ຮູບໜ້າຟອມການຈອງ

### 4.9 ໜ້າຟອມການແຈ້ງອອກ

ເມື່ອເຮົາເຂົ້າມາໃນໜ້າຫຼັກຂອງໂປຣແກຣມແລ້ວຖ້າເລືອກຫນ້າຟອມການແຈ້ງອອກເຮົາກໍ່ຈະເຫັນລາຍລະອຽດດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.9 ຮູບໜ້າຟອມການແຈ້ງອອກ

### 4.10 ໜ້າຟອມຄົ້ນຫາລູກຄ້າ

ເມື່ອເຮົາເຂົ້າມາໃນໜ້າຫຼັກຂອງໂປຣແກຣມແລ້ວຖ້າເລືອກຫນ້າຟອມຄົ້ນຫາລູກຄ້າເຮົາກໍ່ຈະເຫັນລາຍລະອຽດດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.10 ຮູບໜ້າຟອມຄົ້ນຫາລູກຄ້າ

### 4.11 ໜ້າຟອມຄົ້ນຫາພະນັກງານ

ເມື່ອເຮົາເຂົ້າມາໃນໜ້າຫຼັກຂອງໂປຣແກຣມແລ້ວຖ້າເລືອກຫນ້າຟອມຄົ້ນຫາພະນັກງານເຮົາກໍ່ຈະເຫັນລາຍລະອຽດດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 3.11 ຮູບໜ້າຟອມຄົ້ນຫາພະນັກງານ

### 4.12 ໜ້າຟອມຄົ້ນຫາ ຂໍ້ມູນຫ້ອງ

ເມື່ອເຮົາເຂົ້າມາໃນໜ້າຫຼັກຂອງໂປຣແກຣມແລ້ວຖ້າເລືອກຫນ້າຟອມຄົ້ນຫາຂໍ້ມູນຫ້ອງພັກເຮົາກໍ່ຈະເຫັນລາຍລະອຽດດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.12 ຮູບໜ້າຟອມຄົ້ນຫາຂໍ້ມູນຫ້ອງພັກ

### 4.13 ໜ້າຟອມຄົ້ນຫາ ການແຈ້ງອອກ

ເມື່ອເຮົາເຂົ້າມາໃນໜ້າຫຼັກຂອງໂປຣແກຣມແລ້ວຖ້າເລືອກຫນ້າຟອມຄົ້ນຫາຂໍ້ມູນການແຈ້ງອອກເຮົາກໍ່ຈະເຫັນລາຍລະອຽດດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.13 ຮູບໜ້າຟອມຄົ້ນຫາຂໍ້ມູນການແຈ້ງອອກ

### 4.14 ໜ້າຟອມການລາຍງານ

ເມື່ອເຮົາເລືອກໄປທີ່ການລາຍງານ ເຮົາກໍ່ຈະເຫັນລາຍລະອຽດຂອງການລາຍງານກໍ່ຈະມີການຕັ້ງຄ່າ, ຈັດການຂໍ້ມູນ, ການບໍລິການ, ຄົ້ນຫາ, ພິມລາຍງານ.

ຮູບທີ 4.14 ຮູບໜ້າຟອມການລາຍງານ

1. **ການລາຍງານລູກຄ້າ**

ເມື່ອເຮົາເລືອກໄປທີ່ການລາຍງານລູກຄ້າເຮົາກໍ່ຈະເຫັນລາຍລະອຽດຂອງການລາຍງານດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.15 ຮູບໜ້າຟອມການລາຍງານລູກຄ້າ

* **ການລາຍງານການຈອງ**

ເມື່ອເຮົາເລືອກໄປທີ່ການລາຍງານການຈອງ ເຮົາກໍ່ຈະເຫັນລາຍລະອຽດຂອງການລາຍງານດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.16 ຮູບໜ້າຟອມການລາຍງານການຈອງ

1. **ການລາຍງານການເຂົ້າພັກ**

ເມື່ອເຮົາເລືອກໄປທີ່ການລາຍງານເຂົ້າພັກເຮົາກໍ່ຈະເຫັນລາຍລະອຽດຂອງການລາຍງານດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.17 ຮູບໜ້າຟອມການລາຍງານການຈອງ

1. **ການລາຍງານການແຈ້ງອອກ**

ເມື່ອເຮົາເລືອກໄປທີ່ການລາຍງານການແຈ້ງອອກເຮົາກໍ່ຈະເຫັນລາຍລະອຽດຂອງການລາຍງານດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.18 ຮູບໜ້າຟອມການລາຍງານການແຈ້ງອອກ

1. **ການລາຍງານລາຍຮັບວັນ/ເດືອນ/ປີ**

ເມື່ອເຮົາເລືອກໄປທີ່ການລາຍງານລາຍຮັບວັນ/ເດືອນ/ປີເຮົາກໍ່ຈະເຫັນລາຍລະອຽດຂອງການລາຍງານດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.19 ຮູບໜ້າຟອມການລາຍງານລາຍຮັບວັນ/ເດືອນ/ປີ

1. **ການລາຍງານຫ້ອງພັກ**

ເມື່ອເຮົາເລືອກໄປທີ່ການລາຍງານຫ້ອງພັກເຮົາກໍ່ຈະເຫັນລາຍລະອຽດຂອງການລາຍງານດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.20 ຮູບໜ້າຟອມການລາຍງານຫ້ອງພັກ

1. **ການລາຍງານປະເພດຫ້ອງພັກ**

ເມື່ອເຮົາເລືອກໄປທີ່ການລາຍງານປະເພດຫ້ອງພັກເຮົາກໍ່ຈະເຫັນລາຍລະອຽດຂອງການລາຍງານດັ່ງຮູບລຸ່ມນີ້:

ຮູບທີ 4.21 ຮູບໜ້າຟອມການລາຍງານຫ້ອງພັກ

1. **ໃບບິນ**

ຮູບທີ 4. 22 ຮູບໃບບິນ

# **ບົດທີ 5**

# **ສະຫຼຸບ ແລະ ຂໍ້ສະເໜີ**

## 5.1 ສະຫຼຸບ

ລະບົບຄຸ້ມຄອງການຮຽນ-ການສອນຂອງໂຮງຮຽນພອນສະຫັວນ ແລະຊົນເຜົ່າກຽມມະຫາວິທະຍາໄລແມ່ນຖືກພົດທະນາຂຶ້ນເພື່ອຊ່ວຍ ໃຫ້ໂຮງຮຽນມີລະບົບທີ່ທັນສະໄໝສະດວກສະບາຍ ແລະ ເຮັດ ໃຫ້ການຈັດການຂໍ້ມູນ, ການບໍລິການພາຍໃນ ໂຮງຮຽນມີຄວາມວ່ອງໄວ, ມີຄວາມເປັນລະບຽບ, ສະດວກໃນການລາຍງານ, ຂໍ້ມູນທີ່ມີຄວາມຖືກຕ້ອງ ແລະ ຊັດເຈນ. ຂອບເຂດຂອງການດໍາເນີນວຽກງານໂປຣແກຣມ ຂອງພວກຂ້າພະເຈົ້າມີຈັດການຂໍ້ມູນ , ສະໝັກສະມາຊິກ , ບໍລິການ ແລະ ລາຍງານ. ເຊິ່ງເປັນການທົດແທນການເຮັດວຽກໃນລະບົບເກົ່າໂດຍລະບົບໃໝ່ເພື່ອຫຼຸດຜ່ອນບັນຫາການເສຍຫາຍຂອງຂໍ້ມູນ. ພວກນ້ອງສາມາດພັດທະນາໂປຣແກຮມ ເພື່ອຊ່ວຍໃຫ້ເຮັດວຽກສະດວກສະບາຍຍິ່ງຂື້ນ ແລະ ໄດ້ອອກແບບໜ້າຟອມການປ້ອນຂໍ້ມູນຕ່າງໆ.

- ຟອມຈັດການຂໍ້ມູນຫຼັກ 3 ຟອມ

- ສ້າງຟອມຈັດການໄດ້ 5 ຟອມ

- ຟອມການຄົ້ນຫາໄດ້ 5 ຟອມ

- ພິມລາຍງານທັງໝົດໄດ້ 9 ລາຍງານ

## 5.2 ຈຸດດີ

- ຮູ້ວິເຄາະຫາບັນຫາ ແລະ ສາເຫດຂອງລະບົບເກົ່າ

- ຈະໄດ້ລະບົບຈອງປີ້ລົດເມສາຍໃຕ້ແບບອອນໄລໃໝ່

- ສາມາດນໍາໃຊ້ເວບໄຊທີ່ສ້າງຂຶ້ນມາເຂົ້າຊ່ວຍໃນການຈອງປີ້ລົດເມແບບອອນລາຍ

- ເວບໄຊທີ່ສ້າງຂຶ້ນມາສາມາດຈອງອອນລາຍໄດ້

- ການເຮັດບົດລາຍງານສະດວກສະບາຍ ແລະ ວ່ອງໄວຂື້ນ

## 5.3 ຈຸດອ່ອນ

ໂປແກມນີ້ຍັງບໍ່ສົມບູນ ແລະ ຄົບຖ້ວນຕາມຄວາມຕ້ອງການຂອງລະບົບຄຸ້ມຄອງການຮຽນ-ການສອນຂອງໂຮງຮຽນພອນສະຫັວນ ແລະຊົນເຜົ່າກຽມມະຫາວິທະຍາໄລເທື່ອ ເນື່ອງຈາກວ່າໃນການສຶກສາຍັງບໍ່ທັນມີປະສົບການໃນການຂຽນເວບໄຊມາກ່ອນ,ຍັງບໍ່ຄວບຄຸມເຖິງຄວາມຕ້ອງການຕົວຈິງຂອງຜູ້ໃຊ້ລະບົບ.

## 5.4 ແນວທາງໃນການພັດທະນາ ແລະ ຂະຫຍາຍຕໍ່ຂອງສະຖານີ

ເນື່ອງຈາກວ່າໂປແກມນີ້ເປັນໂປຣແກຣມທີ່ສ້າງຂື້ນມາແລ້ວ ດັ່ງນັ້ນເພື່ອເປັນການກວດສວບຫາຂໍ້ພິດພາດ ແລະ ເພື່ອຫາຈຸດດີຈຸດອ່ອນມາທຳການປັບປຸງ ແກ້ໄຂຈື່ງຄວນນຳເອົາໂປຣແກຣມນີ້ໄປປັບປຸງຈຸດບົກພ່ອງຂອງໂປແກຣມໃນບາງສ່ວນ ເພື່ອເຮັດໃຫ້ໂປຣແກຣມມີປະສິດທິພາບ ແລະ ເຮັດວຽກໄດ້ດີຂື້ນ ເພື່ອຈະນຳເອົາໄປປະກອບສ່ວນເຂົ້າໃນການເຮັດວຽກງານຕົວຈິງ

ດັ່ງນັ້ນໂປຣແກຣມນີ້ເປັນໂປຣແກຣມໜື່ງທີ່ມີຄຸນສົມບັດໃນການໃຊ້ຖານຂໍ້ມູນໄດ້.

**ເອກະສານອ້າງອີງ**

**ປະຫວັດຫຍໍ້ຜູ້ຂຽນ**

ຊື່ ແລະ ນາມສະກຸນ: ນາງ ມີໄຊ ສີສຸລິດ

ວັນ,ເດືອນ, ປີເກີດ: 17 ເດືອນ ຕຸລາ ປີ 1989

ບ້ານເກິດ: ພູເຂົາຄຳ ເມືອງ: ແກ້ວອຸດົມ ແຂວງ: ວຽງຈັນ

ບ້ານຢູ່ປັດຈຸບັນ: ໂຊກຄຳ ເມືອງ: ໄຊເສດຖາແຂວງ:ນະຄອນຫຼວງວຽງຈັນ

ການສຶກສາ: ປີ 2013 ຈົບຊັ້ນສູງ ທີ່ ສະຖາບັນໄປສະນີໂທລະຄົມມະນາຄົມ ແລະ ການສື່ສານ

ປີ 2007 ຈົບມັດທະຍົມຕອນປາຍ ທ່າຊັນ

ປີ 2004 ຈົບມັດທະຍົມຕອນຕົ້ນ ພູເຂົາຄຳ

ປີ 2000 ຈົບປະຖົມສົມບູນ ພູເຂົາຄຳ

ເບີໂທ: 020 54787068

ອີເມວ: mexay2020@gmail.com

**ປະຫວັດຫຍໍ້ຜູ້ຂຽນ**

ຊື່ ແລະ ນາມສະກຸນ: ທ້າວ ພອນຄຳ ແກ້ວມະນີ

ວັນ,ເດືອນ,ປີເກີດ: 07 ເດືອນ ມີນາ ປີ 1995

ບ້ານເກິດ:ບ້ານ ທ້ຽວ, ເມືອງ: ວຽງພູຄາ, ແຂວງ: ຫຼວງນໍ້າທາ ບ້ານຢູ່ປັດຈຸບັນ:ບ້ານ ທ້ຽວ, ເມືອງ: ວຽງພູຄາ, ແຂວງ: ຫຼວງນໍ້າທາ

ການສຶກສາ: ປີ 2016 ຈົບຊັ້ນສູງ ທີ່ ສະຖາບັນພັດທະນາສີມືແຮງງານລາວ - ເກົາຫຼີ

ປີ 2013 ຈົບມັດທະຍົມສົມບູນເມືອງວຽງພູຄາ

ປີ 2007 ຈົບປະຖົມສົມບູນ ທີ ໂຮງຮຽນປະຖົມສົມບູນບ້ານດົງວຽງ

ເບີໂທ: 020 2802 2677

ອີເມວ: [phonekham.dev](mailto:lakitptt@gmail.com)@gmail.com