

ສາທາລະລັດ ປະຊາທິປະໄຕ ປະຊາຊົນລາວ

ສັນຕິພາບ ເອກະລາດ ປະຊາທິປະໄຕ ເອກະພາບ ວັດທະນາຖາວອນ

**\*\*\*\*\*\*🙤🗴🙦\*\*\*\*\*\***



ມະຫາວິທະຍາໄລແຫງ່ຊາດ

ຄະນະວິທະຍາສາດທາໍມະຊາດ

ພາກວິຊາວິທະຍາສາດ

ຄອມພິວເຕີ

**ບົດລາຍງານ ວິຊາວິທີການຄົ້ນຄວ້າ**

**ສາຂາ ການພັດທະນາເວບໄຊ**

**ຊື່ຫົວຂໍ້(Tltle)**

ພາສາລາວ: ລະບົບຂາຍ ຮ້ານຂາຍເຄື່ອງແມ່ໄໝ

ພາສາລາວອັງກິດ: Sale Of Maymai Shop

**ສະໜັກຊິໃນກຸ່ມ(Project Team)**

|  |  |  |  |
| --- | --- | --- | --- |
| ລຳດັບ | ລະຫັດນັກສຶກສາ | ຊື່ ແລະ ນາມສະກຸນ | ເບີໂທ |
| 1 | 205N0075/19 | ທ.ຕູ້ຢ່າງທໍ່ຕູ້ ຈົ່ງສືຢ່າງ | 020 .… …. |
| 2 | 205N0083/19 | ທ. ເຊັງວ່າງ ບຼົ່ງໄມ | 02078781525 |

**ອາຈານສອນໂດຍ: ສົນມະນີ ລູຊະວົງ**

**ສົກຮຽນ 2021 – 2022**

1. **ຄວາມເປັນມາ ແລະ ຄວາມສຳຄັນຂອງບັນຫາ**

ໃນປະຈຸບັນການນຳໃຊ້ເຕັກໂນໂລຊີແມ່ນໄດ້ມີອິດທິພົນຫຼາຍໆດ້ານເຊັ່ນ: ດ້ານການສຶກສາ,ດ້ານກຳສະກຳ,ດ້ານການອຸດສະຫາກຳ ແລະ ດ້ານບໍລິການຈະເຫັນໄດ້ວ່າເຕັກໂນໂລຊີກໍໄດ້ມີບົດບາດເຂົ້າມາຫຼາຍໃນການແກ້ໄຂບັນຫາແຕ່ລະອົງກອນ.ໃນທາງດ້ານບໍລິການເອງເຕັກໂນໂລຊີກໍໄດ້ນຳມາແກ້ໄຂບັນຫາຕ່າງໆໃນການວຽກງານໃນການເກັບກຳຂໍ້ມູນເພື່ອຄວາມປອດໄພ,ປ້ອງກັນຄວາມເສຍຫາຍຂອງຂໍ້ມູນ ແລະ ອຳນວຍຄວາມສະດວກໃນການຈັດ ແລະ ຄົ້ນຫາຂໍ້ມູນ.

ຮ້ານຂາຍເຄື່ອງແມ່ໄໝແຟຊັນໃດ້ສ້າງຕັ້ງຂື້ນໃນເມືື່ຶຶຶຶອວັນທີ່ 23 ເດື່ອນ ເມສາ ປີ 2016 ທີ່ບ້ານດົງໂດກ ເມືອງໄຊທານີ ນະຄອນຫຼວງວຽງຈັນເຊິ່ງພາຍໃນຮ້ານຍັງບໍ່ມີລະບົບການຂາຍເຄື່ອງເຂົ້າມາມາຊ່ວຍຈົນເຮັດໃຫ້ຂະບວນການເຮັດວຽກຂອງຮ້ານຍັງບໍ່ມີປະສິດທິພາບ ແລະ ຍັງບໍ່ຕອບສະໜອງຕາມຄວາມຕ້ອງການຂອງເຈົ້າຮ້ານໄດ້ດີເທົ່າທີ່ຄວນເຊັ່ນ: ການເຮັດວຽກຍັງບໍ່ທັນງາ່ຍ,ບໍ່ສະດວກ,ການຄົ້ນຫາລາຍການສີນຄ້າຍັງຊັກຊ້າ,ການຈັດການຂໍ້ມູນພື້ນຖານຍັງບໍ່ທັນເປັນລະບົບລະບຽບ ແລະ ການລາຍງານຍອດຂາຍຍັງບໍ່ທັນວ່ອງໄວ,ບໍ່ຖືກຕ້ອງ ແລະ ຊັດເຈນພໍເຊິ່ງມີຜົນກະທົບໂດຍກົງຕໍ່ການໄດ້ກຳໄລຂອງຮ້ານ ແລະ ການສ້າງໃບບິນຈ່າຍເງີນຍັງເຮັດດ້ວຍມືຈື່ງຊັກຊ້າເສຍເວລາຫຫຼາຍໂດຍເປັນກະທົບຄວາມເພິ່ງພໍໃຈຂອງລູກຄ້າຕໍ່ການບໍລິການຂອງຮ້ານຄ້າ.ດ້ວຍເຫດຜົນເຫຼົ່ານັ້ນພວກຂ້າພະເຈົ້າຈື່ງມີຈຸດປະສົງເພື່ອປະຍຸກໃຊ້ຄວາມສາມາດທາງດ້ານເຕັກໂນໂລຊິຄອມພິວເຕີເຂົ້າມາຊ່ວຍເພື່ອເພີ່ມປະສິດທີພາບໃນການຈັດການຂໍ້ມູນຕ່າງໃນຮູບແບບຂໍ້ມູນດີຈິຕອນ,ການຄົ້ນຫາສິນຄ້າໃຫ້ໄວຂື້ນ,ການພິມໃບບິນຈ່າຍເງີນທີ່ມີຄວາມຖຶກຕ້ອງ ແລະ ວອ່ງໄວ,ສະຫຼຸບລາຍງານສີນຄ້າ,ລາຍງານຍອດຂາຍໃຫ້ຖຶກຕ້ອງ,ຊັດເຈນ ແລະ ຍັງພ້ອມທັງຊ່ວຍໃຫ້ການເຮັດວຽກພາຍໃນຮ້ານສະດວກສະບາຍ ແລະ ວອ່ງໄວຍິ່ງຂື້ນກ່ວາເກົ່າ.

ດັ່ງນັ້ນ,ພວກຂ້າພະເຈົ້າຈື່ງເກີດມີແນວຄວາມຄິດຢາກພັດທະນາລະບົບຂາຍເຄື່ອງນີ້ໃຫ້ ຮ້ານຂາຍເຄື່ອງແມ່ໄໜແຟຊັນໂດຍລະບົບ web Application ເພື່ອແບ່ງເບົາພາລະ ແລະ ຫຼຸດຜ່ອນການໃຊ້ເວລາໃນການເຮັດວຽກຂອງພະນັກງານ,ຊ່ວຍແກ້ໄຂບັນຫາຕ່າງຄື: ຈັດການຂໍ້ມູນພື້ນຖານ,ຈັດການການຂາຍສິນຄ້າ, ຈັດການນຳສິນຄ້າ,ລາຍງານຍອດຂາຍ,ລາຍການສິນຄ້າ , ການຄົ້ນຫາ ແລະ ການລົງທະບຽນຂອງລູນຄ້າ ເຊິ່ງນຳໃຊ້ Vue Js + Laravel ມາຈັດການຂໍ້ມູນ real time,ໂດຍນຳໄຊ້ຖາມຂໍ້ມູນ MySql ມາຈັດເກັບຂໍ້ມູນຕ່າງໆຂອງຮ້ານຄ້າ.

1. **ຈຸດປະສົງ**

ຫຼຸດຜ່ອນຄວາມຜິດພາດໃນການເຮັດວຽກຂອງຮ້ານຂາຍເຄື່ອງແມ່ໄໝແຟຊັນ,ບໍ່ວ່າຈະເປັນການເກັບກຳຂໍ້ ມູນການຂາຍແລະຄິດໄລ່ລາຍໄດ້ຈາກການຂາຍແຕ່ລະມື້, ບັນທຶກລາຍລະອຽດລາຍການສັ່ງຊື້ ແລະ ການນຳເຂົ້າ.ຈຸດປະສົງຫຼັກຂອງການຄົ້ນຄວ້າມີດັ່ງນີ:

* ເພື່ອ ສຶກສາສະພາບ ແລະ ບັນຫາລະບົບ ຂາຍເຄື່ອງແມ່ໄໝແຟຊັນ.
* ເພື່ອພັດທະນາລະບົບການຂາຍ ຂອງ ຮ້ານຂາຍເຄື່ອງ ແມ່ໄໝແຟຊັນ.
* ເພື່ອການຊ່ວຍຈັດເກັດຂໍ້ມູນໄດ້ງ່າຍ,ສະດວກໃນການຄົ້ນຫາ,ຄິດໄລ່ໃບບິນ,ກວດສອບ.
* ເພື່ອການສະຫຼຸບລາຍຮັບ ແລະ ລາຍຈ່າຍ.

1. **ຂອບເຂດ**

ລະບົບຂາຍເຄື່ອງຈະເປັນໃນຮູບແບບຂອງ Client-Server ປະກອບດ້ວຍໜ້າວຽກຫຼັກດັ່ງນີ້:

1. **ຈັດການຂໍ້ມູນພື້ນຖານ**
2. **ຂາຍສິນຄ້າ**
3. **ສັ່ງຊື້**
4. **ນຳເຂົ້າ**
5. **ລາຍງານ**
6. **6)ໂປຼໂມຊັນ**
7. **ປະໂຫຍດທີ່ຄາດວ່າຈະໄດ້ຮັບ**

* ໄດ້ລະບົບການຂາຍຂອງຮ້ານຂາຍເຄື່ອງ ແມ່ໄໝແຟຊັນ.
* ໄດ້ຮັບຄວາມສະດວກໃນການຊ່ວຍຈັດເກັບຂໍ້ມູນໄດ້ງ່າຍ,ສະດວກໃນການຄົ້ນຫາ,ຄິດໄລ່ໃບບິນ,ກວດສອບ.
* ໄດ້ຮັບຄວາມຖືກຕ້ອງ ແລະ ວ່ອງໄວໃນການສະຫຼຸບລາຍຮັບ ແລະ ລາຍຈ່າຍ.

1. **ທົບທວນທິດສະດີ ແລະ ບົດໂຄງການຈົບຊັ້ນທີ່ກ່ຽວຂ້ອງ**
   1. **ທົບທວນທິດສະດີທີ່ກ່ຽວຂ້ອງ**

ລະບົບຂາຍ ຮ້ານຂາຍເຄື່ອງແມ່ໃໝແຟຊັນ ແມ່ນພັດທະນາຂຶ້ນໂດຍນາໍໃຊ້ທິດສະດີຫຼືຄວາມຮູ້ຈາກ 3 ສ່ວນຄື:ທິດສະດິໃນການວິເຄາະ ແລະ ອອກແບບລະບບ, ທີດສະດີກ່ຽວກບັການສາ້ງຖານຂໍ້ມຸນ ແລະຄວາມຮູ້ກ່ຽວກັບພາສາໃນການພັດທະນາລະບົບ. ລາຍລະອຽດກ່ຽວກັບແຕ່ລະທິດສະດີຈະໄດ້ນໍາສະເໜີໂດຍສັງເຂບລຸ່ມນີ້

**5.1.1. ທດິສະດີກ່ຽວກບັການວິເຄາະ ແລະ ອອກແບບລະບົບ**

ການວເິຄາະ ແລະ ອອກແບບລະບບົ (System Analysis and Design) ແມ່ນວິທິທີ່ໃຊ້ໃນການສາ້ງລະບົບຂໍ້ມູນຂ່າວສານໃໜ່ຂຶ້ນມາ ນອກຈາກການສາ້ງລະບບົຂໍ້ມູນຂ່າວສານໃໜ່ຂຶ້ນມາແລ້ວການວິເຄາະລະບົບຍັງຊ່ວຍໃນການແກ້ໄຂບັນຫາຈາກລະບົບຂໍ້ມູນຂ່າວສານເດີມທີ່ມີຢູ່ ແລະ ວເິຄາະຫາຄວາມຕອ້ງການຂອງລະບົບເດີມວ່າຕອ້ງການຫຍັງ. ສ່ວນການອອກແບບລະບົບແມ່ນການນາໍ ເອາົຄວາມຕອ້ງການຂອງລະບົບຂໍ້ມູນຂ່າວສານມາເປັນແບບແຜນ ໃນການພດັທະນາລະບົບໃຫ້ໃຊ້ງານໄດ້ຈິງ (ສົມມິດ ທຸມມະລິ ແລະ ອາມອນ ຈັນທະພາວົງ, 2012).

**5.1.1.1. ວງົຈອນໃນການພດັທະນາລະບົບ**

ວງົຈອນການພດັທະນາລະບບົ ຫຼືSystem Development Life Cycle (SDLC) ເປັນຂະບວນການທີ່ສະແດງເຖງິການດໍາເນນີຂັນ້ຕອນການເຮດັວຽກ ຂອງລະບບົຕັ້ງແຕ່ຕົ໊ນຈົນຈົບມີຂອບເຂດການເຮັດວຽກທີ່ມີໂຄງສ້າງ ແລະ ການຈັດກິດຈະກາໍ ແຕ່ລະໄລຍະຢ່າງຊັດເຈນບົດລາຍງານນີ້ພວກຂ້າພະເຈົ້າຈາກໄດ້ນາໍໃຊ້ທິດສະດີການວິເຄາະ ແລະ ອອກແບບລະບບົ ເຊິ໋ງປະກອບມີ 5 ໄລຍະຄື:

1) ໄລຍະທີ 1 ການວາງແຜນໂຄງການ: ເປັນຂະບວນການພື້ນຖານຂອງຄວາມເຂົ້າໃຈເຖິງຈຸດປະສົງ, ຄວາມຄຸ້ມຄ່າກັບການລົງທືນ ແລະ ຕອ້ງກາໍນີດທີມງານທີ່ຈະມາດໍາເນີນການສາ້ງລະບົບ. ສິ່ງທີຕອ້ງດໍາເນີນໃນໄລຍະນີ້ມີຄື:

* ກາໍນົດບັນຫາ.
* ກາໍນົດເວລາໂຄງການ.
* ສຶກສາຄວາມເປັນໄປໄດ້ຂອງໂຄງການ.
* ຈັດຕັ້ງທີມງານໂຄງການ.
* ດໍາເນີນໂຄງການ.

2) ໄລຍະທີ2 ການວິເຄາະ: ເປັນໄລຍະທີ່ຕອ້ງໄດ້ວິເຄາະວ່າໃຜເປັນຜູ້ໃຊ້ລະບົບຕອ້ງໄດ້ເຮັດຫຍັງແດ່ ແລະ ເຮັດເມື່ອໃດຢູ່ໃສ ພອ້ມທັງລະບຸແນວທາງໃນການປັບປຸງຂະບວນການໃຫ້ດີຂຶ້ນ, ສິ່ງສໍາຄັນຄືຕ້ອງຮວບຮວມຈາກການສັງເກດ, ການສໍາພາດການສ້າງແບບສອບຖາມ,ເອກະສານທີ່ກ່ຽວຂອ້ງ ພອ້ມທັງລະບຽບການຕ່າງໆ. ສິ່ງທີ່ຕອ້ງດໍາເນີນໃນໄລຍະນີ້ມີຄື:

* ວິເຄາະລະບົບງານໃນປັດຈຸບັນ.
* ຮວບຮວມຄວາມຕອ້ງການໃນທຸກໆດາ້ນ, ວິເຄາະ ແລະ ສະຫຼຸບເປັນຂໍ້ກາໍນົດທີ່ຊັດເຈນ.
* ນາໍ ເອົາຂໍ້ກາໍນົດການພັດທະນາມາເປັນຄວາມຕອ້ງການຂອງລະບົບໃໜ່ .
* ສາ້ງແບບຈາໍລອງຂະບວນການຂອງລະບົບໃໜ່ໂດຍການສາ້ງແຜນວາດການໄຫຼຂໍ້ມູນ(DFD).
* ສາ້ງແບບຈາໍລອງຂໍ້ມູນດວ້ຍແຜນວາດຄວາມສໍາພັນລະຫວ່າງຂໍ້ມູນ (ERD).

3) ໄລຍະທີ3 ການອອກແບບ: ເປັນໄລຍະການຕດັສນິໃຈວ່າຈະໃຫລ້ ະບບົດໍາເນນີງານໄປແບບໃດເຊັ່ນ: ການຈັດຫາອຸປະກອນ, ໂຄງສາ້ງຂອງເຄືອຂ່າຍທີ່ຈະນາໍມາໃຊ້,ການຕິດຕໍ່ສື່ສານລະຫວ່າງຜູ້ໃຊ້ກັບລະບົບ , ໂປຣແກຣມຖານຂໍ້ມູນ, ແຟ້ມຂໍ້ມູນ ລວມໄປເຖິງແບບຟອມ ແລະ ການລາຍງານຕ່າງໆ. ສິ້ງທີ່ຕອ້ງດໍາເນນີໃນໄລຍະນີມີຄື:

* ການຈັດຫາລະບົບ .
* ອອກແບບສະຖາປັດຕະຍະກໍາ ຂອງລະບົບ(Architecture Design).
* ອອກແບບຖານຂໍ້ມູນ (Database Design).
* ອອກແບບຟອມລາຍງານ (Output Design).
* ອອກແບບຟອມປອ້ນຂໍ້ມູນ (Input Design).
* ອອກແບບໜ້າຕ່າງຜູ້ໃຊ້(User Interface Design).
* ສ້າງຕົ້ນແບບ (Prototype).
* ອອກແບບໂປຣແກຣມ (Structure Chart).

4) ໄລຍະທີ4 ການນາໍ ໄປໃຊ້ເປັນໄລຍະໃນການສ້າງ, ທດົສອບ ແລະ ຕິດຕັ້ງລະບບົໂດຍມີຈຸດປະສົງຫຼັກໃຫ້ຜູ້ໃ້ຊລະບົບທຸກໆຄົນ ຕອ້ງໄດ້ຜ່ານການເຝິກອົບຮົມການໃຊ້ງານເພື່ອກຽມຄວາມພອ້ມ ຕໍ່ການໃຊ້ລະບົບຂ່າວສານໃຫເ້ກດີປະໂຫຍດຕໍ່ອົງກອນ. ສິ່ງທີ່ຕອ້ງດໍາເນີນໃນໄລຍະນີ້ມີ ຄື:

* ສາ້ງລະບົບຂຶ້ນມາດວ້ຍການຂຽນໂປຣແກຣມ.
* ກວດສອບຄວາມຖຶກຕ້ອງ ແລະ ທົດສອບລະບບົ .
* ແປງຂໍ້ມູນ.
* ຕິດຕັ້ງລະບົບ ແລະ ສ້າງຄູ່ມືລະບົບ .
* ເຝິກອົບຮົມໃຫ້ຜູ້ໃຊ້ແລະປະເມີນຜົນລະບົບໃໝ່ .

1. ໄລຍະທີ 5 ການບໍາລຸງຮັກສາ : ໂດຍທົ່ວໄປແລ້ວໄລະຍະນີ້ບໍ່ໄດ້ຖືກບັນຈຸໃນຂັ້ນຕອນSDLC ເນື່ອງຈາກວ່າເປັນໄລຍະທີໃຊ້ເວລາຫຼາຍທີ່ສຸດຖ້
2. າທຽບກັບໄລຍະອື່ນໆ ເພາະວ່າຕອ້ງໄດ້ບໍາລຸງຮັກສາໃຫ້ລະບົບສາມາດໃຊ້ງານໄດ້ຍາວນານ ແລະ ຮອງຮັບ ເຕັກໂນໂລຊີໃໝ່ ໃນອານາຄົດົ. ສິ່ງທີ່ຕອ້ງດໍາເນີນໃນໄລຍະນີ້ມີ ຄື:

* ການບໍາລຸງຮັກສາ.
* ການເພີ່ມເຕີມ ຄຸນສົມບັດໃຫມ່ເຂົ້າໄປໃນລະບົບ .
* ການສະໜັບສະໜູວຽກຂອງຜູ້ໃຊ້.

**5.1.1.2. ແຜ່ນວາດການໄຫຼຂໍ້ມູນ (Data Flow Diagram: DFD)**

ແຜນວາດການໄຫຼຂໍ້ມູນ ເປັນແບບຈຳລອງຂະບວບການທີ່ຖືກນຳມາໃຊ້ກັບວິທີການພັດທະນາລະບົບຕາມແນວທາງການວິເຄາະ ແລະ ອອກແບບລະບົບໂຄງສ້າງ,ໂດຍແຜນວາດດັ່ງກ່າວນີ້ໃຊ້ເປັນເຄື່ອງມືໃນການພັດທະນາລະບົບແລະ ສະແດງຄວາມສຳພັນລະຫວ່າງຂະບວນການ ແລະ ຂໍ້ມຸນທີ່ກຽ່ວຂ້ອງ,ເຊິ່ງຂໍ້ມຸນໃນແຜນວາດຈະເຮັດໃຫ້ຮູ້ເຖີງຂໍ້ມູນມາຈາກໃສ,ຂໍ້ມຸນໄປທາງໃດ ແລະ ເກິດເຫດການໃດກັບຂໍ້ມູນໃນລະຫ່ວາງການໄຫຼ.

1. ຈຸດປະສົງຂອງການສ້າງແຜດວາດການໄຫຼຂໍ້ມຸນ:

* ເປັນແຜນວາດທີ່ສະຫຼຸບລວມຂໍ້ມູນທັງໜົດທີ່ໄດ້ຈາກການວິເຄາະໃນລັກສະນະຂອງຮູບແບບທີ່ເປັນໂຄງສ້າງ.
* ເປັນຂໍ້ຕົກລົງຮ່ວມກັນລະຫວ່າງນັກວິເຄາະລະບົບ ແລະ ຜູ້ໃຊ້ງານ.
* ເປັນແຜນການທີ່ໃຊ້ໃນການພັດທະນາຕໍ່ໃນຂັ້ນຕອນຂອງການອອກແບບລະບົບ.
* ເປັນແຜນການທີ່ໃຊ້ໃນການອ້າງອີງ ຫຼື ເພື່ອໃຊ້ໃນການພັດທະນາຕໍ່ໃນອານາຄົດ.
* ຮູ້ທີ່ມາໄປຂອງຂໍ້ມູນທີ່ໄຫຼໄປໃນຂະບວນການຕ່າງໆ(Data & Process).

1. ຂັ້ນຕອນການວິເຄະເພື່ອສ້າງແຜນວາດການໄຫຼຂໍ້ມູນ:

* ວິເຄາະໃຫ້ໄດ້ວ່າລະບົບຄວນປະກອບມີ External Entity ໃດແນ່ບໍ່ວ່າຈະເປັນບຸກຄົນ,ໜ່ວຍງານ ຫຼື ລະບົບງານຕ່າງໆ.
* ດຳເນີນການຂຽນແຜນວາດ ທີ່ສະແດງເຖີງສະພາບແວດລ້ອມໂດຍລວມຂອງລະບົບ(Context Diagram).
* ວິເຄາະຂະບົບວ່າຄວນມີຂໍ້ມູນ (Data Store) ໃດແນ່.
* ວິເຄາະຂະບວບການ ຫຼື Process ໃນລະບົບວ່າ ຄວາມມີ process ຫຼັກໆໃດແນ່,ປະກອບມີ Process ຍ່ອຍໃດແນ່.
* ດຳເນີນການຂຽນແຜນວາດການໄຫຼຂໍ້ມູນລະດັບທີ່ 1 ແລະ ອາດຈະມີລະດັບທີ 2 ໃນກໍລະນີທີ່ຈຳເປັນຕ້ອງຂະຫຍາຍລາຍລະອຽດ,ສ່ວນແຜນວາດການໄຫຼຂໍ້ມູນລະດັບທີ່3

ຈະຂຽນຫຼື ບໍ່ຂຽນກໍໄດ້ຕາມຄວາມຕ້ອງການ.

* ການກວດສອບຄວາມສົນດຸຂອງການແຜນນວາດ ແລະ ດັດແກ້ຈົນກວ່າຈະໄດ້ແຜນວາດໄຫຼຂໍ໊ມູນທີ່ສົມບູນ ແລະ ຖືກຕ້ອງ.
* ໃນການສ້າງແຜນວາດການໄຫຼຂໍ້ມູນ ອາດໃຊ້ເຄື່ອງມືຊ່ວຍແຕ້ມເຊັ່ນ ໂປຣແກມ MS Visio ຫຼື ໂປຣແກຣມຊ່ວຍແຕ້ມອື່ນໆກໍໄດ້.

**5.1.1.3. ສັນຍາລັກທີ່ໃຊ້ໃນແຜນວາດການໄຫຼຂໍ້ມູນ**

ຕາຕະລາງທີ່1: ຄວາມໜາຍ ແລະ ສັນຍາລັກໃນແຜນວາດການໄຫຼຂໍ້ມູນ

|  |  |  |
| --- | --- | --- |
| ສັນຍາລັກ | ຊື່ | ຄວາມໝາຍ |
|  | Process | ປະມວນຜົນ ຫຼື  ໜ້າວຽກທີ່ຈະເຮັດໃນ  ໂຄງການນັ້ນໆ |
|  | Data Store | ບ່ອນຈັດເກັບຂໍ້ມູນ |
|  | Boundary  ຫຼື  External Entity | ພາກສ່ວນທີ່ກ່ຽວຂອ້ງ  ກັບລະບົບເຊີ່ງ  ບໍ່ສາມາດຄວບຄຸມໄດ້ |
|  | Data Flow | ການໄຫຼຂອງຂໍ້ມູນ |
|  | Real-Time Link | ການເຊື່ອມໂຍງໄລຍະ ໄກທີ່ມີການຕອບກັບ  ແບບທັນທີ ທັນໃດ |

ຕາຕະລາງທີ່2 : ສັນຍາລັກຂອງ DeMarco & ແລະ Yourdon Gane & Sarson

|  |  |  |  |
| --- | --- | --- | --- |
| DeMarco &  Yourdon | Gane & Sarson | ຊື່ | ຄວາມໜາຍ |
|  |  | Process | ປະມວນຜົນ ຫຼື ໜ້າວຽກ  ທີ່ຈະເຮັດໃນໂຄງການນັ້ນໆ |
|  |  | Data Store | ບອ່ນຈັດເກັບຂໍ້ມູນ |
|  |  | Boundaryຫຼື External  Entity | ພາກສ່ວນທີ່ກຽ່ວຂ້ອງກັບລະບົບເຊີ່ງບໍ່ສາມາດ  ຄຸມໄດ້ |

**5.1.1.4. ຫຼັກການຂຽນແຜນວາດການໄຫຼຂໍ້ມູນ**

ການຂຽນແຜນວາດການໄຫຼຂໍ້ມູນຈະໃຊ້ຫຼັກການຂອງການຂຽນແບບໂຄງສ້າງແຕ່ເທີງລົງລຸ່ມ ຫຼື ຈາກລະບົບໃຫ່ຍໄປຫາລະບົບຍ່ອຍ,ຜູ້ສ້າງແຜນວາດຈຈະຕ້ອງຮູ້ຈັກExternal Entity ທີ່ກຽ່ວຂ້ອງກັບລະບົບທັງໜົດ,ລັກສະນະການເຄື່ອນໄຫຼວຂອງຂໍ້ມູນ ແລະ ການຈັດເກັບຂໍ້ມູນຂອງລະບົບຕ່າງໆ.

ຕາຕະລາງທີ່3 : ຫຼັກການໃນການແຕ້ມແຜນວາດການໄຫຼຂໍ້ມູນ

|  |  |
| --- | --- |
| ອະນຸຍາດ | ບໍອະນຸຍາດ |
| External Entity ສາມາດພົວພັນກັບ Process | External Entity ບ່ໍສາມາດພົວພັນກັບ External Entity |
| Process ສາມາດພົວພັນກັບ External Entity | External Entity ບໍ່ສາມາດພົວພັນກັບ Data Store |
| Process ສາມາດພົວພັນກັບ Process | Data Store ບໍ່ສາມາດພົວພັນກັບ External Entity |
| Process ສາມາດພົວພັນກັບ Data Store | Data Store ບ່ໍສາມາດພົວພັນກັບ Data Store |
| Data Store ສາມາດພົວພັນກັບ Process |  |

1. ຫຼັກການການໃຊ້ສັນຍາລັກ(Process)

* ຂໍ້ມູນບໍ່ພຽງແຕ່ເຂົ້າສູ່ Process ພຽງຢ່າງດຽວໂດຍບໍ່ມີການສົ່ງຂໍ້ມູນອອກຈາກ Process ຈະກໍໃຫ້ເກີດມີຂໍ້ຜິດພາດທີ່ເອີ້ນວ່າ “Black Hole” ເນື່ອງຈາກຂໍ້ມູນມາແລ້ວສູນຫາຍໄປ.
* ຕ້ອງບໍ່ມີແຕ່ຂໍ້ມູນອອກຈາກ Process ພຽງຢ່າງດຽວໂດຍທີ່ບໍ່ມີຂໍ້ມູນເຂົ້າສູ່ Process.
* ຕ້ອງການມີຂໍ້ມູນພຽງພໍທີ່ຈະສສົ່ງຂໍ້ມູນອອກ ເຮັດໃຫ້ເກີດຂໍ້ຜິດພາດທີ່ເອີ້ນວ່າ”Gray Hole”ໂດຍອາດຈະເກີດຈາກການໃຊ້ຊື່ຂໍ້ມູນຮັບເຂົ້າ ແລະ ສົ່ງຂໍ້ມູນອອກມາຜິດ ຫຼື ບໍ່ສົມບຸນ.
* ການຕັ້ງຊື່ Process ຕ້ອງໃຊ້ຄຳກິລິຍາເຊັ້ນ: ຈັດການສີນຄ້າ,ສັ່ງຊື້ສສີນຄ້າ,ກວດສອບສີນຄ້າເປັນຕົ້ນ.

1. ຫຼັກການການໃຊ້ສັນຍາລັກລູກສອນ(Data Flow).

* ຊື່ຂອງ Data Flow ຄວນເປັນຊື່ຂອງຂໍ້ມູນທີ່ສົ່ງໂດຍບໍ່ຕ້ອງອະທີບາຍວ່າສົ່ງແນວໃດເຮັດວຽກແນວໃດ
* Data Flow ຕ້ອງມີຈຸດເລີ່ມຕົ້ນ ຫຼື ຈຸດສີ້ນສຸດທີ່ Process ແລະ Data Flow ແມ່ນຂໍ້ມູນນຳເຂົ້າ ແລະ ສົ່ງອອກຂອງ Process.
* Data Flow ຈະຕ້ອງມີການພົວພັນລະຫວ່າງExternal ກັບ External Entity ບໍ່ໄດ້.
* Data Flow ຈະມີການພົວພັນລະຫວ່າງ External Entity ໄປຫາ Data Flow ບໍ່ໄດ້.
* Data Flow ຈະມີການພົວພັນລະຫວ່າງ Data Store ກັບ External Entity ບໍ່ໄດ້.
* Data Flow ຈະມີການພົວພັນລະຫວ່າງ Data Store ກັບ Data Store ໄດ້.
* ການຕັ້ງຊື່ Data Flow ຕ້ອງເປັນຄຳນານເຊັ່ນ: ຂໍ້ມູນສີນຄ້າທີ່ຜ່ານການກວດສອບຂໍ້ມູນຜູ້ສະໜອງທີ່ຜ່ານການຈັດການເປັນຕົ້ນ.

1. ຫຼັກການຂອງຜູ້ທີ່ກໍ່ໃໃຫ້ເກີດມີຂໍ້ມູນ (External Entity).

* ຂໍ້ມູນຈາກ External Entity ຈະໄປຫາອິກໜື່ງ External Entity ໂດຍກົງບໍ່ໄດ້ຈະຕ້ອງຜ່ານ Process ກ່ອນເພື່ອປະມວນຜົນຂໍ້ມູນນັ້ນນຈື່ໄດ້ຂໍ້ມູນໄປສູ່ອິກໜື່ງ External Entity.
* ການຕັ້ງຈາກ External Entity ຕ້ງໃຊ້ເປັນຄຳນານເຊັ້ນ: ເຈົ້າຂອງຮ້ານ, ຜູ້ສະໜອງເປັນຕົ້ນ.

1. ຫຼັກການການຈັດເກັບຂໍ້ມູນ (Data Store).

* ຂໍ້ມູນຈາກ Data Store ໜື່ງຈະໄປສູ່ອິກໜື່ງ Data Store ໂດຍກົງບໍ່ໄດ້ ຈະຕ້ອງຜ່ານການປະມວນຜົນຈາກ Process ເສ່ຍກ່ອນ.
* ຕັ້ງຊື່ Data Store ຕ້ອງໃຊ້ເປັນຄຳນານເຊັ້ນ: ຂໍ້ມູນລູກຄ້າ,ຂໍ້ມູນຜູ້ສະໜອງເປັນຕົ້ນ.

**5.1.1.5 Flowchart**

Flowchart ແມ່ນແຜນຜັງງ ຫຼື ສັນຍາລັກທີ່ໃຊ້ສສຳຫຼັບອະທີບາຍເຖິງລຳດັບຂັ້ນຕອນ ແລະ ວີທີການເຮັດຂອງວຽກຂອງຂະບວກການໃດໜື່ງ. Flowchart ຖືກໃຊ້ໃນການອອກແບບເພື່ອຊ່ວຍໃຫ້ເຫັນພາບສີ່ງທີ່ເກີດຂື້ນ ແລະ ຊ່ວຍໃນການຫາຂໍ້ຜິດພາດພາບໃນຂະບວນການເຮັດວຽກໄດ້ອີກດ້ວຍ.

ຕາຕະລາງທີ4: ສສັນຍາ ແລະ ຄວາມໜາຍໃນການແຕ້ມ Flowchart

|  |  |
| --- | --- |
|  | ຂະບວນການ, ການຄໍານວນ |
|  | ຮັບຂໍ້ມູນ ຫຼື ສະແດງຂໍ້ມູນໂດຍບໍ່ລະບຸຊະນິດອຸປະກອນ |
|  | ການຕັດສິນໃຈ ຫຼື ການປຽບທຽບ |
|  | ຈຸດເລີ່ມຕົ້ນ ຫຼື ຈຸດສີ່້ນສຸດ |
|  | ສະແດງຜົນທາງເຄື່ອງພີມ |
|  | ປ້ອນຂໍ້ມູນຜ່ານແປ້ນພີມ |
|  | ຈຸດເຊື່ອມຕໍ່ |
|  | ສະແດງຜົນທາງໜ້າຈໍ |

**5.1.2. ທິດສະດີກຽ່ວກັບລະບົບຖາມຂໍ້ມູນ**

ລະບົບຖານຂໍ້ມູນແມ່ນ ການຈັດເກັບຂໍ້ມູນທີ່ມີຄວາມສຳພັນກັນໄວ້ນຳກັນຢ່າງເປັນລະບົບໜາຍຄວາມວ່າແມ່ນການຈັດເກັບຂໍ້ມູນໄວ້ສວ່ນກາງເພື່ອຫຼຸດຜ່ອນຄວາມຊໍ້າຊ້ອນຂອງຂໍ້ມຸນໂດຍຜູ້ໃຊ້ສາມາດເອື້ນໃຊ້ ແລະ ປະຕິບັດກັບຂໍ້ມູນໃນຖານຂໍ້ມູນຮ່ວນກັນໄດ້,ເຊີ່ງຜູ້ໃຊ້ແຕ່ລະຄົນຈະເບີ່ງຂໍ້ມູນໃນມຸມມອງທີ່ແຕກຕ່າງກກັນໄປຕາມຈຸດປະຕາມຈຸດປະສົງຂອງການນຳໃຊ້ (ສົມມິດ ທຸມມາລີ ແລະ ກົງໃຈ ສີສຸຣາດ, 2013).

**5.1.2.1. ການເຮັດ Normalization**

**Normalization** ເປັນວິທີການເພື່ອໃຊ້ເຂົ້າໃນການວິເຄາະ ແລະ ຈັດໂຄງສ້າງຂອງຖານຂໍ້ມູນໃໜ່ໂດຍພະຍາຍາມ ຫຼຸດຄວາມຊໍ້າຊ້ອນຂອງໂຄງສ້າງຖາມຂໍ້ມູນ ເພື່ອໃໃຫ້ໄດ້ໂຄງສ້າງທີ່ມີປະສິດທິພາບ ແລະ ສະດວກໃນເວລາເອົາໄປໃຊ້. ເຊີ່ງວິທີປະຕິບັດແມ່ນຈະເປັນໂຄງສ້າງຂອງຖາມຂໍ້ມູນໃຫ້ຢູ່ໃນຮູບແບບNormalization ໃນລະດັບຕ່າງໆເຊໍັ້ນ: 1NF, 2NF,3NF,BCNF,4NF ແລະ 5NF.

1. Normalization ລະດັບ (1st **Normal Form: 1NF**)

Relation ໜື່ງຈະຢູ່ໃນຮູບ 1NF ກໍຕໍ່ເມື່ອ Attributeໃດໃນ Relation ນັ້ນມີຄ່າໄດ້ຫຼາຍຄ່າ (Multi Valued) ຫຼື ໃນ Relation ນັ້ນບໍ່ມີ Columns ທີ່ມີຄຸນສົນບັດດຽ່ວກັນ (Repeating group).

1. Normal Form ລະດັບ2 (2nd Normal Form: 2NF)

Relation ໜື່ງຈະໃນຮູບ 2NF ກໍ່ຕໍ່ເມືອ Relation ດັ່ງກ່າວເປັນ 1NF ແລະທຸກຄ່າຂອງAttribute ທີ່ບໍ່ແມ່ນສ່ວນປະກອບຂອງຄື (None Key Attribute) ຕ້ອງມີຟັງຊັນການຂື້ນຕໍ່ກັນຂງຄືຫຼັກຢ່າງສົມບຸນ.

1. Normal Form ລະດັບ3 (3rd Normal Form 3NF)

Relation ໜື່ງຈະຢູ່ໃນຮູບ 3NF ກໍຕໍ່ເມື່ອ Relation ດັ່ງດັ່ງກ່າວຢູ່ໃນຮູບ 2NF (Transitive Dependent) ກັບຄືຫຼັກ.

1. Boyee Codd Normal Form (BCNF)

Relation ໜື່ງຢູ່ໃນຮູບແບບ BCNF ກໍຕໍ່ເມື່ຶອ Relation 3NF ແລະ ຕົວເລືອກ (Determinant) ຈະຕ້ອງເປັນ Candidate Key.

1. Normal Form ລະດັບ4 (4nd Normal Form: 4NF)

Relaltion ໜື່ງຈະຢູ່ໃນຮູບ 4NF ກໍຕໍ່ເມື່ອ Relaltion ດັ່ງກ່າວໃນຮູບແບບ 3NF ຫຼື BCNF ແລະ ບໍ່ມີການຂັ້ນຕໍ່ກັນແບບກຸ່ນໃນRelaltion.

1. Normal Form ລະດັບ 5(5nd Normal Form 5NF)

Relaltion ໜື່ງຈະຢູ່ໃນຮູບ 5NF ກໍຕໍ່ເມື່ອບໍ່ມີ Cyclic Dependency ເຊິ່ງຈະເກີດຂື້ນກັບ Relaltion ທີ່ມີຄ່າຄີຫຼັກປະກອບດວ້ຍ Columnsຫຼື Attribute ຕັ້ງແຕ່3 ຄ່າຂື້ນໄປ.

**5.1.2.2. ແຜນວາດຄວາມສຳພັນລະຫ່ວາງ(ER Diagram)**

ER Diagram (Entity Relationship Diagram) ແມ່ນແຜນຜັງສະແດງຄວາມສຳພັນລະຫວ່າງຂໍ້ມູນເຊີ່ງປະກອບດ້ວຍ: Entity, Attribute, Relationship.

1. ຄວາມໜາຍຂອງ: Entity ໜາຍເຖິງຂໍ້ມູນທີ່ເຮົາສົນໃຈເຊີ່ງອາດເປັນສີ່ງທີ່ສາມາດເບີ່ງເຫັນ,ຈັບ ແລະ ສຳຜັດໄດ້ເຊັ່ນ:ຄົນ,ສັດ,ພະນັກງານເປັນຕົ້ນ ຫຼື ອາດເປັນສີ່ງທີ່ມີລັກສະນະຂອງມະໂນພາບເຊັ່ນ: ອາຊິບ ຫຼື ລາຍວິຊາທີ່ຕ້ອງລົງທະບຽນຮຽນ. ສັນຍາລັກທີ່ໃຊ້ແທນ Entity ແມ່ນຮູບສີ່ແຈສາກດ້ານໃນບັນຈຸດ້ວຍໃນບັນຈຸດ້ວຍຊື່ຂອງ Entity ສຳຫຼັບ Entityທີ່ຂື້ນກັບ Entity ອື່ນເອີ້ນວ່າ: Weak Entity ມີສັນຍາລັກເປັນຮູບສີ່ແຈສາກແຕ່ມີສອງເສັ້ນດັ່ງຕາຕະລາງລຸ່ມນີ້:

ຕາຕະລາງທີ່5 : ຄວາມໜາຍ ແລະ ສັນຍາລັກ Entity

|  |  |  |
| --- | --- | --- |
| Entity | | ແທນ Entity |
| Entity | | ແທນ Weak Entity |
| Composite Entity | ແທນ Entity ເມື່ອເກີດຄວາມສໍາພັນແບບ  ຫຼາຍຕໍ່ຫຼາຍ | |

1. Attribute ເປັນສີ່ງທີ່ບົ່ງບອກເຖິຄຸນລັກສະນະຂອງ Entity ຈະມີຄຸນສົມບັດສະເພາະເຊັ່ນ: Entity ພະນັກງານທີ່ສັງກັດຢູ່ໃນບໍລິສັດ, ສີ່ງທີ່ເປັນຕົວອະທີບາຍສຳຫຼັບພະນັກງານດັ່ງກ່າວແມ່ນ: ຊື່ພະນັກງານ, ອາຍຸ, ທີ່ຢຸ່ ເປັນຕົ້ນສັນຍາລັກທີ່ໃຊ້ແທນ Attribute ຈະໃຊ້ຮູບແອນລິບທີ່ມີເສັ້ນເຊື່ອມໂຍງຈາກ Entity ແລະ ພາຍໃນຮູບແອນລິບຈະບັນທືກຊື່ Attribute.

ຕາຕະລາງທີ6: ຄວາມໜາຍ ແລະ ສັນຍາລັກຂອງ Attribute

|  |  |
| --- | --- |
|  | ແທນ Attribute |
|  | ແທນ Primary key Attribute |
|  | ແທນ Derive Attribute |
| Attribute Name | ແທນ Multi-Valued Attribute |
| Attribute Name  Attribute Name  Attribute Name | ແທນ Composite Attribute |

1. ຄວາມໜາຍຂອງ Relationship

Relationship ແມ່ນສີ່ງທີ່ໃຊ້ສະແດງເຖິງຄວາມສຳພັນລະຫວ່າງ Entity 2 Entity ຫຼື ຫຼາຍກວ່າ 2 Entity, ສຳລັບ Entity ແຕ່ລະຕົວອາດເກີດຄວາມສຳພັນໄດ້ຫຼາຍກວ່າ 1 ຄວາມສຳພັນ. ເຮົາສັນຍາລັກ Relationship ດ້ວຍຮູບດອກຈັນ,ພາຍໃນບັນຈຸຊື່ຄວາມສຳພັນ ແລະ ມີເສັ້ນເຊື່ອມໂຍງໄປຍັງ Entity ທີ່ເກີດຄວາມສຳພັນ.ການແບ່ງປະເພດຄວາມສຳພັນສາມາດຈຳແນກໂດຍໃຊ້ Cardinality Ratio ຫຼື ຈຳນວນນສະມາຊິກທີ່ເກີດຄວາມສຳພັນເຊີ່ງງສາມາດຈັດແບ່ງເປັນ 3 ກຸ່ມດັ່ງນີ້:

* ຄວາມສຳພັນແບບໜື່ງຕໍ່ໜື່ງ(1:1)

ຖ້າ Entity E1 ມີຄວາມສຳພັນກັບ Entity E2 ແບບໜື່ງຕໍ່ໜື່ງນັ້ນໜາຍຄວາມວ່າ ສະມາຊຊິກຂອງ Entity E1 ໜື່ງລາຍການຈະມີຄວາມສຳພັນກັບສະມາຊິກໃນ Entity E1 ໄດ້ໜື່ງລາຍການເຊັ່ນກັນຕົວຢ່າງ:

1

1

1

1

Student

Tel

Tel

Has

H

as

ຮູບທີ່1: ຄວາມສຳພັນແບບ 1 ຕໍ່ 1

* ຄວາມສຳພັນແບບໜື່ງຕໍ່ຫຼາຍ (1: N Relationship)

ຖ້າ Entity E1 ມີຄວາມສຳພັນ Entity E2 ແບບໜື່ງຕໍ່ຫຼາຍ, ນັ້ນໜາຍຄວາມວ່າສະມາຊິກຂອງ Entity E1 ໜື່ງລາຍການຈະມີຄວາມສຳພັນກັບສະມາຊິກໃນ Entity E2 ໄດ້ຫຼາຍກວ່າໜື່ງລາຍການ, ໃນທາງກົງກັນຂ້ານສະມາຊິກຂອງ Entity E2 ໜື່ງລາຍການຈະມີຄວາມສຳພັນນກັບສະມາຊິກໃນ Entity E1 ໄດ້ພຽງໜື່ງລາຍການເທົ່ານັ້ນຕວົຢ່າງ:

M

1

Teach

H

as

Student

Tel

Teacher

ຮູບທີ່2: ຄວາມສຳພັນແບບ1 ຕໍ່ ຫຼາຍ

* ຄວາມສຳພັນແບບຫຼາຍຕໍ່ຫຼາຍ(N:M Relationship)

ຖ້າ Entity E1ມີຄວາມສຳພັນກັບ Entity E2 ແບບຫຼາຍຕໍ່ຫຼາຍ,ນັ້ນໜາຍຄວາມວ່າສະມາຊິກຂອງ Entity E1 ໜື່ງລາຍການຈະມີຄວາມສຳພັນກັບສະຊິກໃນ Entity E2 ໄດ້ຫຼາຍກວ່າໜື່ງລາຍໜື່ງລາຍການ, ໃນທາງກົງກັນຂ້ານສະມາຊິກຂອງ Entity E2 ໜື່ງລາຍການຈະມີຄວາມສຳພັນກັບສະມາຊິກໃນ Entity E1 ຫຼາຍກວ່າໜື່ງລາຍການເຊັ່ນດຽ່ວກັນຕົວຢ່າງ:

M

M

Register

Register

Register

gister

Student

Subject

r

Tel

ຮູບທີ່2: ຄວາມສຳພັນແບບ ຫຼາຍຕໍ່ ຫຼາຍ

**5.1.2.3. ພາສາ SQL(Structure Query Language)**

ພາສາ SQL(Structure Query Language) ຖຸກພັດທະນາໂດຍບໍສັດ IBM(IBM’s San Jose Research Laboratory) ໃນລັດຄາລິຟໍເນຍຊ່ວງຕົ້ນປີ ຄ.ສ 1970. ເປັນພາສາທີ່ໃຊ້ໃນການຈັດການຖາມຂໍ້ມູນເກືອບທຸກຕົວເຊັ່ນນ: SQL, Server, Oracle, Access ໂດຍຮູບແບບຂອງຄຳສັ່ງມາດຕະຖານທີ່ຖືກກຳນົດໂດຍANSI (American National Standards Institute) ໃນປີ ຄ ສ 1986. ມີຮູບແບບຂອງຄຳສັ່ງທີ່ງາຍຕໍ່ການໃຊ້ງານເພາະຄ້າຍພາສາມະນຸດ (“Structure Query Language(SQL)”, 2017).

**5.1.2.4. ລະບົບຈັດການຖາມຂໍ້ມູນ (Database Management Systems: DBMS)**

ລະບົບຈັດການຖາມຂໍ້ມູນ(DBMS)ແມ່ນໂປຣແກຣມທີ່ເຮັດໜ້າທີ່ໃນການບໍລິຫານ ແລະ ຈັດການຖາມຂໍ້ມູນໃນການສ້າງ, ການເອີ້ນໃຊ້, ການແກ້ໄຂ ແລະ ການລຶບ. ປຽບສະເໜືອນຕົວກາງລະຫວ່າງຜູ້ໃຊ້ ກັບ ລະບົບຖາມຂໍ້ມູນນ (“Database Management Systems( DBMS)”,2015), ອົງປະກອບຂອງລະບົບຈັດການຖາມຂໍ້ມູນປະກອບມີ:

* ພົົດຈະນຸກົນຂໍ້ມູນ (Data Dictionary): ມີໜ້າທີ່ອະທິບາຍເຖີງລາຍລະອຽດຂອງໂຄງສ້າງຂໍ້ມູນ.

ຕາຕະລາງທີ່7 : ຕົວຢ່າງນັກຮຽນ

|  |  |  |
| --- | --- | --- |
| StudentID | StudentName | Age |
| 25N0075/19 | ທ.ຕູ້ຢ່າງທໍ່ຕູ້ ຈົ່ງສືຢ່າງ | 21 |
| 25N0083/19 | ທ. ເຊັງວ່າງ ບຼົ່ງໄມ | 22 |

* ເຄຶ່ອງມືອຳນວຍຄວາມສະດວກ(Utility)
* ພາສາລະບົບຈັດຖາມຂໍ້ມູນ(DBMS Language) ປະກອບມີ 3 ພາກສ່ວນຄື:

1. ພາສາສຳຫຼັບກຳນົດໂຄງສ້າງຂອງຖາມຂໍ້ມູນ(Data Definition Language: fDDL)ໃຊ້ໃນນການສ້າງຕາຕະລາງ, ກຳນົດໂຄງສ້າງຂອງຕາຕະລາງ. DDL ປະກອບມີຄຳສັ່ງນີ້:

* CREATE ສຳຫຼັບຕາຕະລາງງ.
* DROP ລຶບຕາຕະລາງ.
* ALTER ແກ້ໄຂໂຄງສ້າງຂອງຕາຕະລາງ.

1. ພາສາສຳຫຼັບກຳນົດໂຄງສ້າງຂອງຖາມຂໍ້ມູນ (Data Manipulation Language: DML) ໃຊ້ສຳຫຼັບເອີ້ນໃຊ້,ເພີ່ມ, ລົບ, ແລະ ແກ້ໄຂຂໍ້ມູນໃນຕາຕະລາງປະກອບມີຄຳສັ່ງດັ່ງນີ້:

* SELLECT ເອີ້ນໃຊ້ຂໍ້ມູນໃນຕາຕະລາງ
* INSERT ເພີ່ມຂໍ້ມູນໃນຕາຕະລາງ
* DELETE ລືບຂໍ້ມູນໃນຕາຕະລາງ
* UPDATE ແກ້ໄຂຂໍ້ມູນໃນຕາຕະລາງ

1. ພາສາທີ່ໃຊ້ຄວບຄຸມຂໍ້ມຸນ(Data Control Language: DCL) ໃຊ້ໃນການກຳນົດສິດອະນຸຍາດ ຫຼື ຍົກເລີກ ການເຂົ້າເຖີງການຂໍ້ມູນເພື່ອປ້ອງກັນຄວາມປອດໄພຂອງຖາມຂໍ້ມູນປະກອບມີຄຳສັ່ງດັ່ງນີ້:

* GRANT ກຳນົດສິດໃນການເຂົ້າເຖີງຂໍໍ້ມູນຂອງຜຜູ້ໃຊ້.
* REVOKE ຍົກເລີກສິດໃນການເຂົ້າເຖີງຂໍໍ້ມູນຂອງຜຜູ້ໃຊ້.
* ເຄື່ອງມືສ້າງລາຍງານ (Report Generator): ແມ່ນໂປຣແກຣມທີ່ສ້າງລາຍງານ ແລະ ສະແດງຂໍ້ມູນທາງຈໍພາບ ຫຼື ເຄື່ອງພິມ.
* ການຮັກສາຄວາມປອດໄພ ໃນການເຂົ້າເຖີງຂໍ້ມູນ(Access Security): ເປັນເຄື່ອງມືທີຊ່ວຍໃໃຫຫ້ຜູ້ດຸແລສາມາດກຳນົດສິດໃນການເຂົ້າເຖີງຂໍ້ມູນ.
* ການກູ້ລະບົບ(System Recovery): ຊ່ວຍໃນການກູ້ຄິນຂໍ້ມູນເມື່ອຖານຂໍ້ມູນເກີດຄວາມເສຍຫາຍ ຫຼື ຂັດຂ້ອງ.

**5.1.3. ພາສາທີ່ໃຊ້ໃນການຂຽນໂປຣແກຣມ**

ພາສາທີ່ໃຊ້ໃນການຂຽນໂປຣແກຣມແມ່ນພາສາປະດິດຊະນິດໜື່ງທີ່ອອກແບບຂື້ນມາເພື່ອສື່ສານກັບຄອມພີວເຕີ.

**5.1.3.1. ທິດສະດີພື້ນຖານກຽ່ວກັບ Visual Studio Code**

Visual Studio Code ເປັນໂປຣແກຣມ Code Editor ໜື່ງທີ່ຊວ່ຍໃນການພັດທະນາ, ແກ້ໄຂ ແລະ ປັບແຕ່ງໂຄ້ດ, ພັດທະນາຂື້ນໂດຍບໍລິສັດ Microsoft. ມີການພັດທະນາອອກມາໃນຮູບແບບຂອງ OpenSource ຈຶ່ງສາມາດນໍາມາໃຊ້ງານໄດ້ແບບຟຣີໆ. Visual Studio Code ນັ້ນ, ຮອງຮັບການໃຊ້ງານໃນ Windows , macOS ແລະ Linux. ຮອງຮັບທັງພາສາ JavaScript, TypeScript ແລະ Node.js. ສາມາດເຊື່ອມຕໍ່ກັບ Git ໄດ້ນໍາມາໃຊ້ງານງ່າຍບໍ່ຊັບຊ້ອນມີເຄື່ອງມືສ່ວນຂະຫຍາຍຕ່າງໆໃຫ້ເລືອກໃຊ້ຢ່າງຫຼາຍເຊັ່ນ: 1.ການເປີດໃຊ້ງານພາສາອື່ນໆເຊັ່ນ: C++, C#, Java, Python, PHP ຫຼື Go 2.Themes, 3.Debugger ແລະ 4.Commands ເປັນຕົ້ນ.

**5.1.3.2. ຄວາມຮູ້ກຽ່ວກັບ. Node js**

**Node.js** ແມ່ນສະພາບແວດລ້ອມການທຳງານຂອງພາສາ JavaScript ຢູ່ນອກ webbrowser ທີ່ທໍາງານດ້ວຍ V8. ນີ້ຫມາຍຄວາມວ່າພວກເຮົາສາມາດນໍາໃຊ້ Node.js ເພື່ອພັດທະນາແອັບພລິເຄຊັນ ແບບ Command line ແອັບພລິເຄຊັນ desktop, ແລະແມ້ກະທັ້ງເຄື່ອງແມ່ຂ່າຍເວັບ ໂດຍທີ່ Node.js ຈະມີ APIs ທີ່ພວກເຮົາສາມາດໃຊ້ສໍາລັບການເຮັດວຽກກັບລະບົບປະຕິບັດການເຊັ່ນ: ການຮັບຄ່າ, ການສະແດງຜົນ, ການອ່ານ​​ຂຽນ​ໄຟລ ແລະ ເຮັດວຽກກັບເຄືອຂ່າຍເປັັນຕົ້ນ.

Node js ຖືກພັດທະນາແລະທໍາງານໂດຍໃຊ້ Chrome V8 engine ສຳລັບຄອມໄພເລີດພາສາ Javascript ໃຫ້ເປັນພາສາເຄື່ອງດ້ວຍການຄອມໄພເລີດແບບ Just-In-Time (JIT) ເພື່ອເພື່ມປະສິທິພາບການທໍາງານຂອງພາສາ Javascript ຈາກທີ່ແຕ່ເດິມມັນເປັນພາສາທີ່ມີການທໍາງານແບບ Interpreted

Node js ເປັນໂປຼແກຣມທີ່ສາມາດໃຊ້ໄດ້ທັ້ງບົນ Windows, Linux ແລະ Mac OS ສະແດງວ່າພວກເຮົາສາມາດຂຽນຄັ້ງດຽວແລະນໍາໄປຣັນໄດ້ທຸກລະບົບປະຕິບັດການ ນີ້ເປັນແນວຄິດຂອງການຂຽນຄັ້ງດຽວແຕ່ທໍາງານໄດ້ທຸກທີ່ (Write once, run anywhere).

Node js ຖືກພັດທະນາຂື້ນມາຄັ້ງທໍາອິດໂດຍນັກພັດທະນາຊາວອາເມລິກັນ Ryan Dahl ໃນປີ 2009 ສໍາລັບໃຊ້ເປັນສະພາບແວດລ້ອມເພື່ອທີ່ຈະໃຊ້ພາສາ JavaScript ໃນການພັດທະນາໃນຮູບແບບ Server Side

**5.1.3. ພາສາ JavaScrit**

ພາສາ JavaScript ເປັນພາສາການຂຽນໂປລແກລມທີ່ທໍາງານໃນຝັ່ງຂອງ Client ຖືກພັດທະນາແລະປະຕິບັດຕາມຂໍ້ກໍານົດມາດຕະຖານ ECMAScript; JavaScript ເປັນພາສາລະດັບສູງ. ຄອມໄພລໃນຂະນາທີ່ໂປຼແກຣນລັນ(JIT) , ພາສາ JavaScript ເປັນເທກໂນໂລຍີຫຼັກຂອງການພັດທະນາເວັບໄຊ (World Wide Web) ຊ່ວຍໃຫ້ຫນ້າເວັບສາມາດພົວພັນກັບຜູ້ໃຊ້.

ພາສາ JavaScript ຖືກອອກແບບແລະສ້າງໂດຍ Brendan Eich ສຳລັບພາສາ script ທີ່ທໍາງານບົນເວັບບຣາວເສີ Navigator, ພາສາ JavaSript ໄດ້ຖືກເປີດຕົວແລະເປັນສ່ວນໜຶ່ງຂອງເວັບບຣາວເສີ Navigator ໃນເດືອນ ກັນຍາ 1995 ໂດຍໃຊ້ຊື່່ວ່າ LiveScript ແລະປ່ຽນເປັນ Javascript ໃນສາມເດືອນຕໍ່ມາ

**5.1.3.4. ຄວາມຮູ້ກຽ່ວກັບ. Express js**

**Express.js** ເປັນ Web Application Framework ສຳລັບທໍາງານບົນ Platform ຂອງ Node js ເຊິ່ງເປັນເສິເວິບຕົວໜຶ່ງ, ສໍາລັບການພັດທະນາ Express js ໃນເວັບຈະເວົ້າເຖິງການໃຊ້ Routing ໃນການກຳນົດເສັ້ນທາງຂອງລະບົບ ເເລະ Middleware ການຮັບສົ່ງຂໍ້ມູນຂອງລະບົບ ໃຊ້ຂຽນໃນຮູບແບບ mvc ໃນການເຊື່ອມຕໍ່ກັບຖ່ານຂໍ້ມູນໃນ MySql

**5.2. ທົບທວນບົດຄົ້ນຄ້ວາທີ່ກ່ຽວຂ້ອງ.**

ບັນຫາການຈັດການດ້ານການຂາຍ ເປັນບັນຫາໜຶ່ງເຊິ່ງໄດ້ຮັບຄວາມສົນໃຈຕໍ່ນັກຄົ້ນຄ້ວາທັງດ້ານສາຍວິທະຍາສາດທໍາມະຊາດ ແລະ ສາຍວິທະຍາສາດສັງຄົມ. ເວົ້າສະເພາະສາຍວິທະຍາສາດທໍາມະຊາດກໍຄື: ສາຂາການພັດທະນາເວັບໄຊ໌ ໄດ້ປະກົດມີຫຼາຍບົດຄົ້ນຄ້ວາທີ່ສຶກສາບັນຫາ ແລະ ນໍາໃຊ້ເທັກໂນໂລຊີເຂົ້າໃນການພັດທະນາລະບົບໃນຮູບແບບໂປຣແກຣມນໍາໃຊ້ ແລະ ໃນຮູບແບບ Website ເພື່ອຊ່ວຍແກ້ໄຂຂໍ້ຫຍຸ້ງຍາກ ແລະ ອໍານວຍຄວາມສະດວກ.“ລະບົບບໍລິຫານການຂາຍເບ້ຍໄມ້ ຮ້ານຈັນຟອງ” ເປັນຫົວຂໍ້ບົດຄົ້ນຄ້ວາໜຶ່ງທີ່ຜູ້ພັດທະນາ (ດາແກ້ວມະນີວົງ ພ້ອມດ້ວຍຄະນະສຶກສາ, 2018) ໄດ້ສຶກສາສະພາບ ແລະ ບັນຫາໃນການຂາຍເບ້ຍໄມ້, ເນື່ອງຈາກພາຍໃນ ຮ້ານຈັນຟອງ ບໍ່ມີໂປຣແກຣມນໍາໃຊ້ສະເພາະໃນການຈັດການຂໍ້ມູນເຊັ່ນ: ການບັນທືກຂໍ້ມູນຊັບພະຍາກອນໃນຮ້ານ, ຂໍ້ມູນການຊື້ສິນຄ້າເຂົ້າຮ້ານ,ຂໍ້ມູນການຂາຍ,ຂໍ້ມູນພະນັກງານ ແລະ ຂໍ້ມູນເບີກຈ່າຍເງິນເດືອນຍັງຈົດກ່າຍໃສ່ເຈ້ຍເຮັດໃຫຂ້ໍມູນສັບສົນປົນເປກັນ ແລະ ບາງຄັ້ງຕົກເຮ່ຍເສຍຫາຍ, ຍາກຕໍ່ການກວດສອບ ແລະ ຄົ້ນຫາ. ລະບົບດັ່ງກ່າວຈະພັດທະນາອອກມາໃນຮູບແບບ Website ໂດຍນຳໃຊ້ My SQL ເປັນຖານຂໍ້ມູນ,ນຳໃຊ້ HTML ເປັນພາສາຫຼັກໃນການພັດທະນາ Website.

Website ປະກອບມີ 6 ໜ້າວຽກຫຼັກຄືຈັດການຂໍ້ມູນພື້ນຖານ, ສັ່ງຊື້ສິນຄ້າເຂົ້າຮາ້ນ, ສະໝັກສະມາຊິກ, ຂາຍສິນຄ້າ, ຄົ້ນຫາຂໍ້ມູນ ແລະ ລາຍງານ. ພາຍຫຼງັ Website ສ້າງສໍາເລັດຜົນໄດ້ຮັບຄາດວ່າຈະສາມາດແກ້ໄຂບັນຫາທີ່ເກີດຂຶ້ນ ແລະ ຕອບສະໜອງຕໍ່ຄວາມຕ້ອງການຂອງຜູ້ໃຊ້ເປັນຢ່າງດີ(ຄໍາພັນ ພິລາວັນ ພ້ອມດ້ວຍກຸ່ມສະມາຊິກຄົ້ນຄວ້າ, 2016).ເປັນອີກກຸ່ມໜຶ່ງທີ່ໃຫ້ຄວາມສົນໃຈຕໍ່ການພັດທະນາລະບົບບໍລິຫານການຂາຍສິນຄ້າ ເຊິ່ງໄດ້ໃສ່ຊື່ຫົວຂໍ້ວ່າ:“ໂປຣແກຣມບໍລິຫານການຂາຍສິນຄ້າ ແລະ ສ້ອມແປງຄອມພິວເຕີຂອງສູນ ເອສເອສ - ຄອມພິວເຕີ” ສູນ SS-Computer ແຫ່ງນີ້ແມ່ນສູນບໍລິການຂາຍອຸປະກອນໄອທີ ແລະ ການສ້ອມແປງຄອມພິວເຕີ. ລະບົບທີ່ພັດທະນາຂຶ້ນມາຈະຢູ່ໃນຮູບແບບ On line ທີ່ປະກອບມີ 7 ໜ້າວຽກຫຼັກຄື: ຈັດການຂໍ້ມຸນຫຼັກ,ຈັດຊື້ສິນຄ້າ,ບໍລິການການຂາຍ,ບໍລິການຫຼັງການຂາຍ,ບໍລິການສ້ອມແປງ,ບໍລິິການຫຼັງການສ້ອມແປງ,ລາຍງານ.

ໂປຣແກຣມຈະອອກມາຮູບແບບໃດ ແລະ ໃຊ້ພາສາຫຍັງໃນການພັດທະນາແມ່ນບໍ່ໄດ້ກ່າວໄວ້ໃນບົດ,ມີພຽງແຕ່ບອກວ່າຈະຈັດເກັບຂໍ້ມູນໂດຍໃຊ້ Microsoft SQL Server 2008 ແລະ ພັດທະນາໂປຣແກຣມໂດຍໃຊ້ Microsoft Visual Studio 2010. ເນື່ອງຈາກປັດຈຸບັນເປັນຍຸກຫັນເຂົ້າເທັກໂນໂລຊີທັນສະໄໝຄວາມຕ້ອງການການນໍາໃຊ້ໂປຣແກຣມເພື່ອການຈັດການດ້ານການຂາຍຍັງມີຫຼາຍບໍ່ວ່າຈະເປັນຮ້ານຂະຫນາດໃຫຍ່ ຫຼື ນ້ອຍ. ໄດ້ເຫັນເຖິງຄວາມສໍາຄັນດັ່ງກ່າວຈຶ່ງມີຄວາມຕັ້ງໃຈທີ່ຈະພັດທະນາ “ລະບົບຈັດການຂາຍເຄືອງສອ້ມແປງຄອມພິວເຕີທົ່ວໄປຮ້ານ ທ້າວຕົ້ນສ້ອມແປງຄອມພິວເຕີ ” ນີ້ຂຶ້ນມາ.

**6 ວີທີດຳເນີນການຄົ້ນຄ້ວາ**

ຂັ້ນຕອນ ແລະ ການດຳເນີນການໂຄງການໂດຍຫຼັກໆແລ້ວພວກຂ້າພະເຈົ້າໄດ້ອີ່່ງໃສ່ຕາມວົງຈອນການພັດທະນາລະບົບຂອງ SDLC (System Development Life Cycle) ດັ່ງລຸ່ມນີ້:

Project Planning

Phase

1

Analysis

Phase

2

Design Phase

3

Imblemantation

Phase

4

Maintenance

Phase

5

SDLC

ຮູບທີ່ 1 : ວົງຈອນການພັດທະນາລະບົບ

**6.1 ໄລຍະການວາງແຜນໂຄງການ**

ໃນໄລຍະວາງແຜນໂຄງການ ເປັນຂັ້ນຕອນການກຳນົດຂອບເຂດຂອງບັນຫາ,ສາເຫດຂອງບັນຫາຈາກດຳເນີດງານໃນປັດຈຸບັນ, ຄວາມເປັນໄປໄດ້ໃນການສ້າງລະບົບໃໜ່. ສີ່ງທີ່ພວກຂ້າພະເຈົ້າດຳເນີນການໄລຍະນີ້ ແມ່ນການສຳຫຼວດຄວາມຕ້ອງການຂອງຜູ້ໃຊ້ລະບົບ. ພວກຂ້າພະເຈົ້າໄດ້ລົງສຳພາດຕົວຈີງຈາກເຈົ້າຂອງຮ້ານ,ສືກສາລະບົບການເຮັດວຽກ ແລະ ລະບົບການຂາຍຂອງ ທ້າວ ຕົ້ນຂາຍເຄື່ອງພິວເຕີທີ່ມີຢູ່ໃນປັດຈຸບັນ.ສຶກສາຄວາມເປັນໄປໄດ້ຂອງລະບົບເພື່ອໃຫ້ໂປຣແກຣມມີປະສິດທພາບ ແລະ ສອດຄ່ອງກັບຄວາມຕ້ອງການ.

**6.2. ໄລຍະການວີເຄາະ**

ເປັນໄລຍະທີ່ພວກເຮົາພວກຂ້າເຈົ້ານຳຄວາມຕ້ອງການຜູ້ໃຊ້ລະບົບທີ່່ໄດ້ຈາກການສຳຫຼວດມາວິເຄາະລະອຽດເພື່ອເປັນຂໍ້ກຳນົດຂອງລະບົບໃໜ່.ໂດຍນຳໄຊ້ວີທີການວິເຄາະ ແລະ ອອກແບບໂຄງສ້າງໂດຍໃຊ້ແບບຈຳລອງຂອງ Process Model ເຊີ່ງປະກອບ: ແຜນວາດການໄຫຼຂໍ້ມູນ(DFD: Data Flow Diagram) ແລະ ແຜນວາດຄວາມສຳພັນລະຫວ່າງຂໍ້ມູນ (ERD: Entity Relationship Diagram). ເພື່ອເຮັດໃຫ້ຮູ້ເຖີງລາຍລະອຽດຂັ້ນຕອນການເຮັດວຽກໃນລະບົບວ່າປະກອບດ້ວຍຫັຍງແດ່,ມີຄວາມສຳພັນ ຫຼື ກຽ່ວຂ້ອງກັນແນວໃດ.

**6.3. ໄລຍະການອອກແບບ**

ໄລຍະການອອກແບບມ່ນໄລຍະທີ່ພວກຂ້າພະເຈົ້ານຳເອົາຜົນທີ່ໄດ້ຈາກການວິເຄາະມາອອກແບບລະບົບໂດຍການອອກແບບໜ້າຟອມຫຼັກ, ໜ້າຟອມປ້ອມຂໍ້ມູນ ແລະ ໜ້າຟອ້ມລາຍງານໂດຍນຳໃຊ້ Visual Studio 2019.ອອກແແບຖາມຂໍ້ມູນວ່າຈະຕ້ອງເກັບກຳລາຍລະອຽດຫັຍງໂດຍນາໃຊ້ MySQL Server , 2019.

**6.4. ໄລຍະການພັດທະນາ ແລະ ຕິດຕັ້ງ**

ໄລຍະການພັດທະນາ ແລະ ຕິດຕັ້ງ ແມ່ນໄລຍະທີ່ດຳເນີນການສ້າງລະບົບດ້ວຍການຂຽນໂປຣແກຣມໂດຍໃຊ້ພາສາVB.NET.ເພື່ອພັດທະນາລະບົບຂື້້ນມາພ້ອມທັງຕິດຕັ້ງລະບົບເພື່ອໃຊ້ງານ.

**6.5. ໄລຍະການທິດສອບ ແລະ ການຳໃຊ້**

ທິດສອບການເຮັດວຽກໄດ້ຈີງ ຫຼື ບໍ່. ເພື່ອນຳມາປັງປຸງ ແລະ ແກ້ໄຂໃຫ້ຖືກຕ້ອງ. ຖ້າເຫັນວ່າມີຄວາມສົມບູນແລ້ວ ຈື່ງນຳເອົາໂປຣແກຮມດັ່ງກ່າວໄປນຳໃຊ້ຈີງ,ພ້ອມທັງສ້າງຄູ່ມືປະກອບກາ່ນໃຊ້ລະບົບ.

1. **ສະຖານທີ້ໃນການຄົ້ນຄ້ວາ**

ສະຖານທີ່ໃນການສຶກສາຄົ້ນຄ້ວາຂອງພວກຂ້າພະເຈົ້າແມ່ນໄດ້ ດໍາເນີນການຄົ້ນຄ້ວາຢູ່ ພາກວິທະຍາສາດຄອມພິວເຕີຄະນະວິທະຍາສາດທໍາມະຊາດ, ມະຫາວິທະຍາໄລແຫ່ງຊາດ ແລະ ໄດ້ລົງເກັບການຂໍ້ມູນຢູ່ທີ່ ຮ້ານທ້າວຕົ້ນສ້ອມແປງຄອມພິວເຕີ, ບ້ານ ດົງໂດກ, ເມືອງໄຊທານີ, ນະຄອນຫຼວງວຽງຈັນ.

1. **ໄລະເວລາໃນການດຳເນີນການຄົ້ນຄວ້າ**

ແຜນການດຳເນີນງານຂອງ **“**ລະບົບຈັດການຂາຍເຄືອງສອ້ມແປງຄອມພິວເຕີທົ່ວໄປຮ້ານ ທ້າວຕົ້ນສ້ອມແປງຄອມພິວເຕີ**”** ໂດຍເລີ່ມການສືກສາຄວາມເປັນໄປໄດ້ໃນການສຶກສາດັ່ງຕາຕະລາງລຸ່ມນີ້:

**ຕາຕະລາງທີ8: ໄລຍະປະຕິບັດ**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ລຳດັບ** | **ໜ້າວຽກທີ່ດຳເນີນງານ** | **ເວລາໃຊ້** | **ໄລຍະເວລາ** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| **2021** | | | | | | | | | | | | | **2022** | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| **ຕູລາ** | | | | **ພະຈິກ** | | | | | **ທັນວາ** | | | | **ມັງກອນ** | | | | **ກຸມພາ** | | | | **ມີນາ** | | | | **ເມສາ** | | | | **ພືດສະພາ** | | | | **ມີຖຸນາ** | | | | **ກໍລະກົດ** | | | |
| **1** | **2** | **3** | **4** | | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** |
| **1** | **ຈັດກຸ່ນແລະເລືອກຫົວຂໍ້** | **1**  **ອາທິດ** |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2** | **ສະເໜີຫົວຂໍ້ແລະສະມາຊິກ** | **1**  **ອາທິດ** |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3** | **ເກັບກຳຂໍ້ມູນ** | **1**  **ອາທິດ** |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4** | **ຂຽນບົດສະເໜີໂຄງການ** | **6**  **ອາທິດ** |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5** | **ສົ່ງບົດສະເໜີໂຄງການ** | **ອາທິດ** |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **6** |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. **ເຄື່ອງມືໃນການຄົ້ວຄ້ວາ**

**9.1 ເຄື່ອງມືທາງດ້ານ Hardware**

ເຄື່ອງມືທາງດ້ານ Hardware ທີ່ໃຊ້ປະກອບໃນການເຮັດວຽກໃນຄັ້ງນີ້ປະກອບມີ:

* Notebook Computer Sony VAIO
* Processor Intel (R) Core (TM) i5-6200U CPU @ 2.30GHZ 2.40 GHZ
* RAM 16 GB
* Operating System 64-bit.
  1. **ເຄື່ອງມືທາງດ້ານ Software**

ເຄື່ອງມືທາງດ້ານ Software ທີ່ໃຊ້ປະກອບໃນການເຮັດວຽກໃນຄັ້ງນີ້ປະກອບມີ:

* Microsoft Windows 10 Professional 64bitG
* Microsoft Visio Studio 2019.
* Microsoft SQL Server 2019.
* Microsoft Office Professional Plus 2019.
* Microsoft Visio 2019.
* Crystal Report.
  1. **ເຄື່ອງມືທີ່ໃຊ້ໃນການເອົາຂໍ້ມູນ**

**ເຄື່ອງມືທີ່ໃຊ້ໃນການເອົາຂໍ້ມູນໃນຄັ້ງນີ້ພວກເຮົາໄດ້ໃຊ້ວິທີໃນການສຳພາດກັບເຈົ້າຂອງຮ້ານປະກອບມີ: ບີກ, ເຈ້ຍ,HUAWEI Assistant Y6P ເພື່ອຈົດບັນທືກຂໍ້ມູນ, ຂັ້ນຕອນໃນການເຮັດວຽກ ແລະ ວິທີການຕ່າງໆ.**

1. **ເອກະສານອ້າງອີງ.**

ສົມມິດ ທຸມມະລີ ແລະ ອາມອນ ຈັນທະພາວົງ.(2012).*ວິເຄາະ ແລະ ອອກແບບລະບົບ*.ນະຄອນຫຼວງວຽງ ຈັນ: ຄະນະວິທະຍາສາດທຳມະຊາດ,ມະຫາວິທະຍາໄລແຫ່ງຊາດ.

ສົມມິດ ທຸມມາລີ ແລະ ກົງໃຈ ສິສຸຣາດ.(2013).*ລະບົບຖານຂໍ້ມູນ.*ນະຄອນຫຼວງວຽງຈັນ: ຄະນະ ວິທະຍາສາດທຳມະຊາດ,ມະຫາວິທະຍາໄລແຫ່ງຊາດ.

ອໍຣະຍາປຮິຊາພານິດ.(ພ. ສ 2557).*ຄູ່ມືຮຽນການວີເຄາະ ແລະ ອອກແບບບົບສະບັບສົມບູນ.*ນົນ

ທະບູລີ:ບໍລິສັດໄອດີຊີພຣິມຽມຈຳກັດ.

ດາ ແກ້ວມະນີວົງພ້ອມດ້ວຍຍຄະນະສືກສາ.(2018).ບົດໂຄງການຈົບຊັ້ນປະລິນຍາຕິວິທະຍາສາດ: *ລະບົບບໍ ຫານການຂາຍເບ້ຍໄມ້ຮ້ານຈັນຟອງ.*

ຄຳພັນ ພີລາວັນ ພ້ອມດ້ວຍຄະຍະສືກສາ.(2016). ບົດໂຄງການຈົບຊັ້ນປະລີນຍາຕິວີທະຍາສາດ:*ໂປຣແກຣມ ບໍລິຫານການຂາຍສີນຄ້າ ແລະ ສ້ອມແປງຄອມພິວເຕີຂອງສສູນ ເອສເອສ - ຄອມພິວເຕີ.*

Structure Query Language (SQL). (2017, ກຸມພາ 18). ໄດ້ຈາກ :