

System Analysis and Design

ໂດຍ: ອຈ ສິມມິດ ທຸມມາລີ

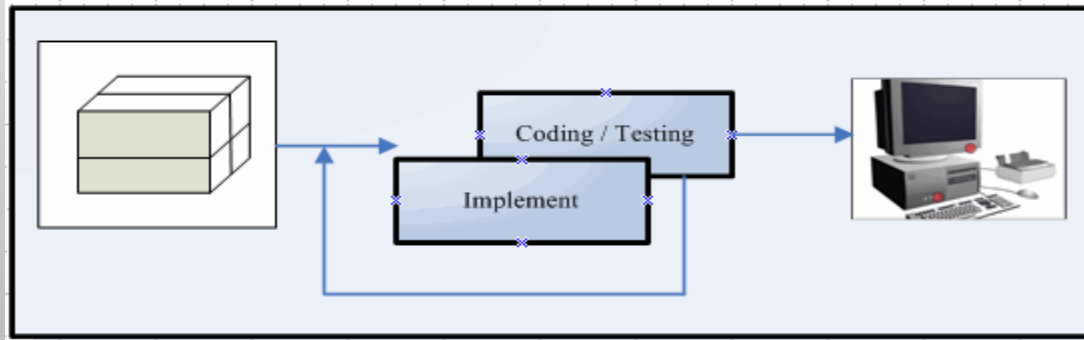
ໄລຍະທີ 4-5 ການນຳໄປໃຊ້ ແລະ ການບຳລຸງຮັກສາ

ເມື່ອການອອກແບບລະບົບໄດ້ດຳເນີນສຳເລັດແລ້ວ ນັກວິເຄາະລະບົບຈະຕ້ອງດຳເນີນການສ້າງລະບົບຕາມໄປຕາມທີ່ໄດ້ອອກແບບໄວ້ ແລະ ພັດທະນາເອກະສານປະກອບລະບົບຂຶ້ນມາ. ໄລຍະນີ້ ນັກຂຽນໂປຣແກຣມຈະໄດ້ໃຊ້ເວລາໃນການຂຽນໂປຣແກຣມ, ໃນຂະນະດຽວກັນ ນັກວິເຄາະລະບົບກໍຈະຈັດທຳແຜນການທົດສອບຂຶ້ນມາກ່ອນການຕິດຕັ້ງໃຊ້ງານຈິງເພື່ອກວດສອບການດຳເນີນງານຂອງລະບົບວ່າໄປຕາມທີ່ຄາດຫວັງໄວ້ຫຼືບໍ່ ແລະ ໃນທີ່ສຸດເອກະສານຕ່າງໆທີ່ສ້າງຂຶ້ນຈາກການອອກແບບຖືກນຳມາພັດທະນາເປັນຮູບປະທຳໃນຮູບແບບຂອງລະບົບງານຈິງ.

ບົດທີ 9 ການພັດທະນາ ແລະ ບຳລຸງຮັກສາ

ການພັດທະນາລະບົບ

ເມື່ອວິເຄາະ ແລະ ອອກແບບລະບົບແລ້ວຂັ້ນຕອນຕໍ່ມາກໍແມ່ນ ການພັດທະນາລະບົບເຊິ່ງຂັ້ນ ຕອນການພັດທະນາລະບົບນີ້ຈະລວມໄປເຖິງ ການຂຽນໂປຣແກຣມ(Coding), ການທົດສອບ(Testing) ແລະ ການ ຕິດຕັ້ງລະບົບ(Implement).



ການຂຽນໂປຣແກຣມ

- ສຶກສາເອກະສານຕ່າງໆ
- ອອກແບບໂປຣແກຣມ
- ຂຽນໂປຣແກຣມ
- ທົດສອບໂປຣແກຣມ
- ສ້າງເອກະສານຄູ່ມືການຂຽນໂປຣແກຣມ

ສຶກສາເອກະສານຕ່າງໆ

ສຶກສາເອກະສານຕ່າງໆໝາຍເຖິງການສຶກສາເອກະສານທີ່ໄດ້ຈາກ
ການວິເຄາະ ແລະ ອອກ ແບບລະບົບຈາກນັກວິເຄາະລະບົບ ເຊິ່ງປະກອບ
ດ້ວຍແຜນວາດການໄຫຼຂໍ້ມູນ, ຄໍາອະທິບາຍການປະມວນຜົນຂໍ້ມູນ,
ພຶດຈະນານຸກົມຂໍ້ມູນ, ຮູບແບບຈໍພາບທີ່ໃຊ້ເພື່ອບັນທຶກ ຫຼື ສະແດງຂໍ້ມູນ
ແລະ ຮູບແບບລາຍງານ ລວມທັງເອກະສານທີ່ເປັນແຫຼ່ງຂໍ້ມູນຕ່າງໆ.
ເອກະສານດັ່ງກ່າວນີ້ຈະເຮັດໃຫ້ນັກຂຽນໂປຣແກຣມມີຄວາມເຂົ້າໃຈໃນ
ລະບົບ ແລະ ຂຽນໂປຣແກຣມໄດ້ຕາມທີ່ຕ້ອງການ.

ອອກແບບໂປຣແກຣມ

ຫຼັງຈາກທີ່ໄດ້ສຶກສາເອກະສານຕ່າງໆແລ້ວ ນັກຂຽນໂປຣແກຣມຈະ
ເລີ່ມອອກແບບໂປຣແກຣມ, ແກ້ບັນຫາ ແລະ ຕັດສິນໃຈເລືອກແນວທາງ
ໃນການພັດທະນາລະບົບ ເພື່ອໃຫ້ການພັດທະນາລະບົບໃນຂັ້ນຕອນນີ້
ປະສົບຜົນສໍາເລັດ ໂດຍອາດຈະໃຊ້ເຄື່ອງມືໃນການພັດທະນາໂປຣແກຣມ
ເຊັ່ນ: ແຜນຜັງໂປຣແກຣມ (Program Flowchart), ການຂຽນລະຫັດ
ຈຳລອງ (Pseudocode) ເພື່ອສະດວກໃນຂັ້ນຕອນຂອງການຂຽນໂປຣ
ແກຣມ.

ຂຽນໂປຣແກຣມ

ການຂຽນໂປຣແກຣມຄື ການລົງມືຂຽນໂປຣແກຣມ ໂດຍມີການພິຈາລະນາພາສາຄອມພິວເຕີທີ່ໄດ້ເລືອກໄວ້ແລ້ວ. ໃນປັດຈຸບັນມີໂປຣແກຣມພ້ອມເຄື່ອງມືທີ່ອ່ານວຍຄວາມສະດວກໃນການຂຽນໂປຣແກຣມຢ່າງຫຼວງຫຼາຍສາມາດເລືອກໃຊ້ຄໍາສັ່ງການເຮັດວຽກໃນ Menu ຂອງໂປຣແກຣມ ແລະ Generate ເປັນລະຫັດຄໍາສັ່ງໄດ້ທັນທີ ເຊັ່ນ ການອອກແບບຈໍພາບ, ເມື່ອອອກແບບເປັນທີ່ຮຽບ ຮ້ອຍແລ້ວໃຫ້ໂປຣແກຣມແປງສິ່ງທີ່ເຮົາອອກແບບເປັນລະຫັດຄໍາສັ່ງ.

ທົດສອບໂປຣແກຣມ

ການທົດສອບໂປຣແກຣມຄືການນຳເອົາໂປຣແກຣມທີ່ຂຽນຂຶ້ນມາທົດສອບຄວາມ
ຖືກຕ້ອງໃນ ຮູບແບບຂອງພາສາ (Syntax) ແລະ ສູດຄຳນວນຕ່າງໆ ທີ່ໃຊ້ວ່າ
ເຮັດວຽກຖືກຕ້ອງຕາມ ຄວາມຕ້ອງການ ຫຼື ບໍ່ ໂດຍສະເພາະແມ່ນສູດທີ່ໃຊ້
ຄຳນວນໃນໂປຣແກຣມນັກຂຽນໂປຣແກຣມອາດ ສ້າງສູດຜິດພາດໄດ້ ຫຼື ມີການ
ຂຽນ Logic ທີ່ຜິດພາດ ເຊິ່ງໃນສ່ວນນີ້ຈະ Compile ຜ່ານ ໂດຍທີ່ຄອມພິວເຕີບໍ່
ສາມາດຮູ້ໄດ້ວ່າເປັນສູດຄຳນວນທີ່ຜິດ. ດັ່ງນັ້ນນັກຂຽນໂປຣແກຣມຕ້ອງມີການ
ກວດສອບຢ່າງຖີ່ຖ້ວນເຊິ່ງອາດມີການສົມມຸດຂໍ້ມູນຂຶ້ນມາ ແລະ ດຳເນີນການ
ປະມວນຜົນປຽບທຽບ ຜົນຮັບທີ່ໄດ້ວ່າ ຖືກຕ້ອງ ຫຼື ບໍ່ ເປັນຕົ້ນ.

ສ້າງເອກະສານຄູ່ມືການຂຽນໂປຣແກຣມ

ການສ້າງເອກະສານຄູ່ມືການຂຽນໂປຣແກຣມເພື່ອປະໂຫຍດໃນການປັບປຸງແກ້ໄຂໂປຣແກຣມໃນອະນາຄົດ ເຊັ່ນວ່າ ອາດມີການປັບແກ້ ຫຼື ມີການເພີ່ມ Module ກໍສາມາດນຳເອົາເອກະສານ ທີ່ໄດ້ສ້າງໄວ້ນັ້ນມາສຶກສາ ແລະ ພັດທະນາໂປຣແກຣມຕໍ່ໄປ.

ການທົດສອບລະບົບ (System Testing)

ການທົດສອບລະບົບເປັນການຮັບປະກັນຄວາມຖືກຕ້ອງໃນສ່ວນຂອງການກວດສອບທາງດ້ານ

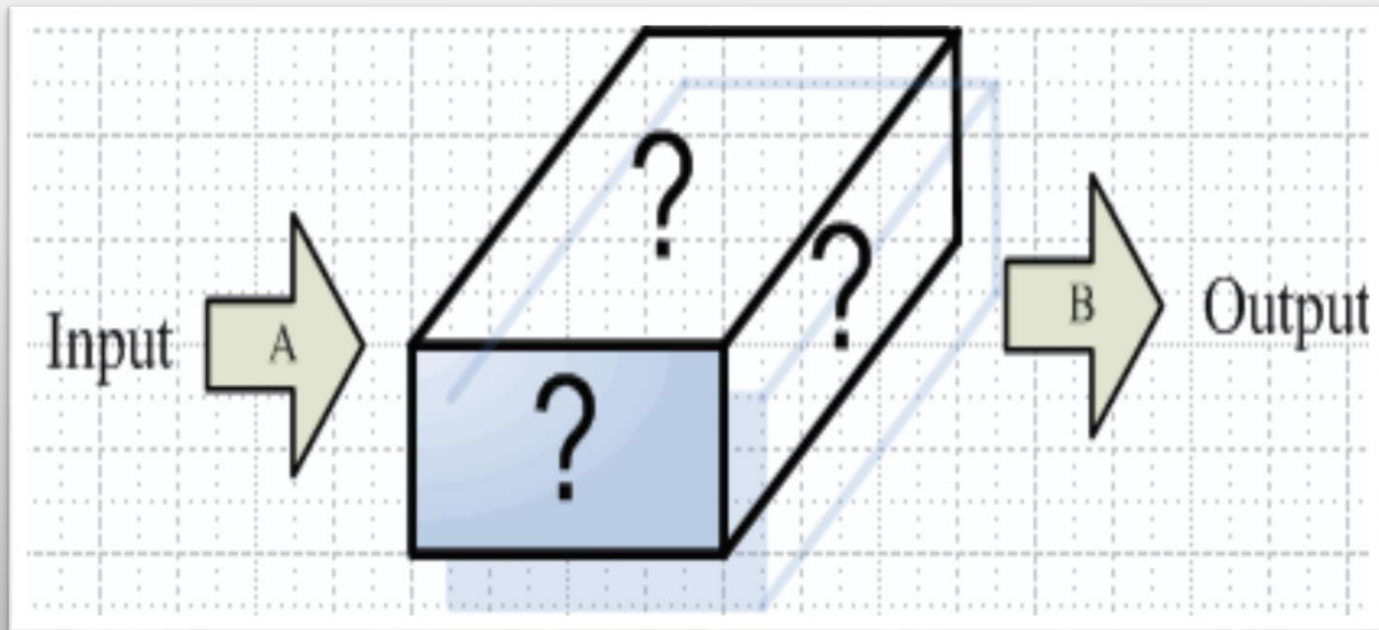
- Verification ຄືການກວດສອບຄວາມຖືກຕ້ອງຫຼັງຈາກຍອມຮັບໃນລາຍລະອຽດຂອງລະບົບ.
- Validation ຄືການກວດສອບຄວາມຖືກຕ້ອງຈາກຄວາມຕ້ອງການຂອງຜູ້ໃຊ້ວຽກ.

ຄວາມຖືກຕ້ອງທາງດ້ານ Verification ສາມາດກວດສອບໄດ້ຫຼັງຈາກການພິຈາລະນາ ລາຍລະອຽດຂອງລະບົບທີ່ຈະດໍາເນີນການພັດທະນາເຖິງວ່າຈະຖືກຕ້ອງໃນລາຍລະອຽດທີ່ກໍານົດໄວ້ແຕ່ ການສົ່ງມອບລະບົບຕ້ອງໄດ້ຜ່ານການກວດສອບທາງດ້ານ Validation, ລູກຄ້າອາດຈະບໍ່ຍອມຮັບໃນລະບົບ ຖ້າບໍ່ກົງກັບຄວາມຕ້ອງການຂອງລູກຄ້າ.

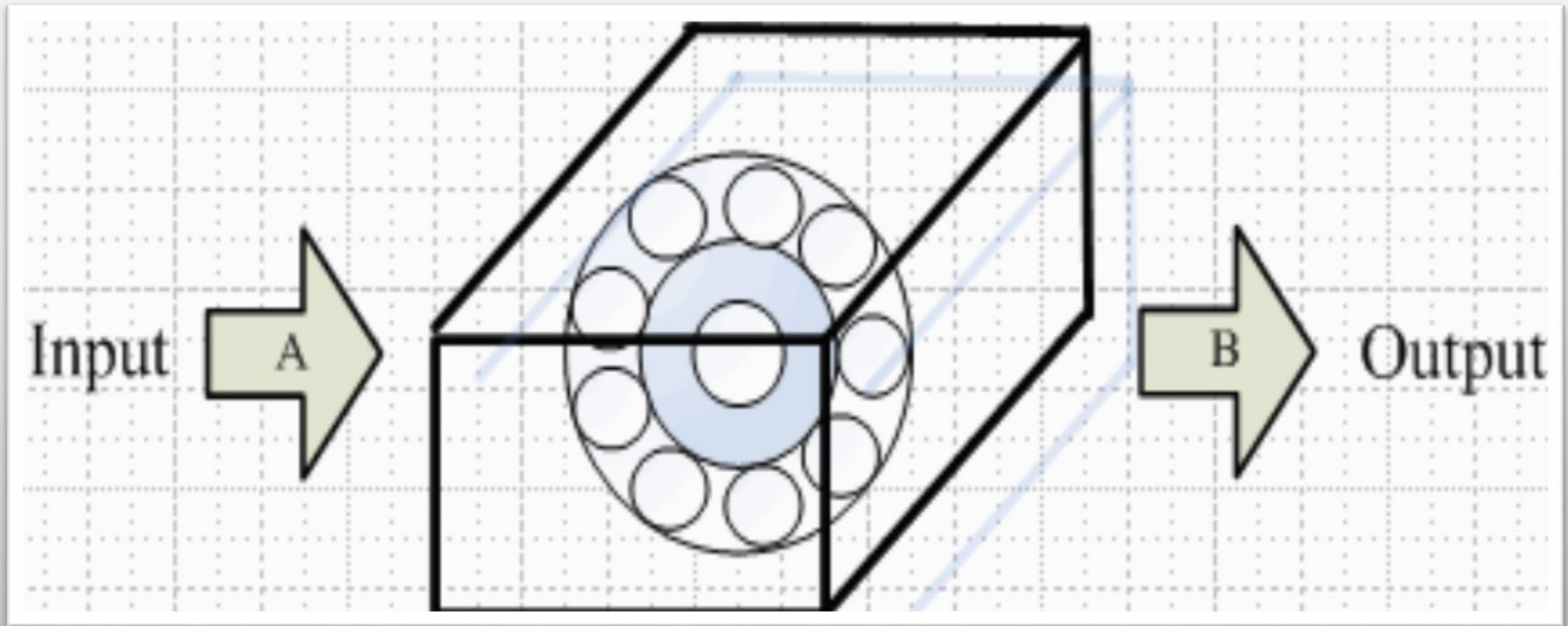
ກິນລະຍຸດໃນການທົດສອບ (Testing Strategies)

- Black Box Testing
- White Box Testing

Black Box Testing



White Box Testing



ການຈັດຫາອຸປະກອນລະບົບ

ຈາກລາຍລະອຽດທີ່ກ່າວມາທັງໝົດຈະເນັ້ນໜັກໃນສ່ວນຂອງ Software ທີ່ດີກົງກັບຄວາມ ຕ້ອງການ ແຕ່ Software ທີ່ດີ ຫຼື ລະບົບວຽກທີ່ດີ ກໍຈະຕ້ອງມີສ່ວນປະກອບຂອງ Hardware ທີ່ໃຊ້ວຽກຮ່ວມກັບ Software ນັ້ນຢ່າງເໝາະສົມ. ດັ່ງນັ້ນແນວທາງໃນການຈັດຫາອຸປະກອນຂອງ ລະບົບ (System Acquisition) ຈຶ່ງເປັນເລື່ອງທີ່ສໍາຄັນສໍາລັບການພິຈາລະນາອຸປະກອນທາງ Hardware ໃຫ້ເໝາະສົມ ກັບ Software ທີ່ໄດ້ມາ.

ໃນການຈັດຫາອຸປະກອນທີ່ຕ້ອງການ ເຊັ່ນ: Hardware, Software ລວມທັງການບໍລິ ການສາມາດຈັດຫາດ້ວຍວິທີຕ່າງໆດັ່ງລຸ່ມນີ້

- ການຈັດຊື້ (Purchase)
- ການເຊົ່າ (Renting)

ການຈັດຊື້ (Purchase)

- ໃຫ້ຜູ້ຂາຍຈັດສັນໃຫ້ທັງໝົດໃນລະບົບ (Turnkey Vendor) ເປັນການຈັດຊື້ທັງລະບົບ ລວມທັງອຸປະກອນ Hardware, Software ການບໍລິການຈາກຜູ້ຂາຍພຽງຜູ້ດຽວລະບົບທີ່ຈັດຊື້ພ້ອມໃຊ້ ວຽກໄດ້ທັນທີ.
- ບອກລາຍລະອຽດຄວາມຕ້ອງການຂອງອຸປະກອນ ແລະ ໃຫ້ຜູ້ຂາຍຫາຈັດຊື້ມາໃຫ້ເປັນການຈັດຊື້ດ້ວຍການບອກຄວາມຕ້ອງການ ເຊິ່ງອາດຈະບອກລາຍລະອຽດໃຫ້ກັບບໍລິສັດຕ່າງໆ ແລະ ໃຫ້ ຕອບກັບມາໃນຮູບຂອງໃບສະເໜີລາຄາປະມານ 3-5 ບໍລິສັດ ແລະ ພິຈາລະນາການປຽບທຽບລາຄາ, ການບອກລາຍລະອຽດຂອງຄວາມຕ້ອງການນີ້ ອາດຈະບໍ່ໄດ້ຄໍານຶງເຖິງຊະນິດ, ຍີ່ຫໍ້, ຂາໜາດທີ່ຕ້ອງ ການ ຫຼື ອາດຈະມີການລະບຸຊະນິດ, ຍີ່ຫໍ້, ຂາໜາດທີ່ຕ້ອງການໄດ້.
- ຮູ້ລາຍລະອຽດຂອງອຸປະກອນຕ່າງໆ ແລະ ຈັດຊື້ເອງເປັນການຈັດຊື້ອຸປະກອນດ້ວຍຕົນເອງ ເຊິ່ງຜູ້ຊື້ຈະຕ້ອງມີຄວາມຮູ້ກ່ຽວກັບລາຍລະອຽດຂອງອຸປະກອນຂ້ອນຂ້າງດີພໍສົມຄວນ.

ການເຊົ່າ (Renting)

ການເຊົ່າເປັນການເຮັດສັນຍາເຊົ່າຕາມໄລຍະເວລາທີ່ສັນຍາ
ຮ່ວມກັນໂດຍມີການເຊົ່າໃນໄລຍະສັ້ນ ແລະ ໄລຍະຍາວ ເຊັ່ນ ການ
ເຊົ່າໃນໄລຍະສັ້ນລະຫວ່າງ 1-12 ເດືອນ ການເຊົ່າໃນໄລຍະຍາວ
ເຊັ່ນ 3-5 ປີ. ເມື່ອຄົບກຳໜົດອາດຍົກເລີກສັນຍາເຊົ່າ ຫຼື ຕ້ອງການ
ເຊົ່າຕໍ່ກໍໄດ້. ການເຊົ່າເໝາະ

ສົມກັບລະບົບວຽກທີ່ຕ້ອງການຄວາມຢືດຢູ່ນກັບລະບົບຄອມພິວເຕີທີ່
ຕ້ອງການໃຊ້ເທັກໂນໂລຊີທີ່ສູງຂຶ້ນຢູ່ສະເໝີ ຫຼື ການມີຕົ້ນທຶນຈຳກັດ
ຫຼື ມີການໃຊ້ວຽກໃນຊ່ວງເວລາໃດໜຶ່ງເປັນຕົ້ນ.

ການຕິດຕັ້ງລະບົບ (Implement)

- ຕິດຕັ້ງເພື່ອໃຊ້ວຽກໃໝ່ໄດ້ທັນທີ (Direct Changeover)
- ການຕິດຕັ້ງລະບົບແບບຄູ່ຂະໜານ (Parallel Running)
- ການຕິດຕັ້ງແບບເທື່ອລະເຟດ (Phase Changeover)
- ການຕິດຕັ້ງແບບເປັນຕົວຢ່າງ (Pilot Project)

ຕິດຕັ້ງເພື່ອໃຊ້ວຽກໃໝ່ໄດ້ທັນທີ (Direct Changeover)

ຄືການຢຸດໃຊ້ວຽກລະບົບເດີມ ແລະ ຫັນມາໃຊ້ລະບົບໃໝ່ທັນທີເປັນການປັບປ່ຽນທີ່ງ່າຍທີ່ສຸດ ແຕ່ກໍມີຂໍ້ເສຍຄື ອາດເກີດບັນຫາຂໍ້ຜິດພາດຈາກລະບົບທີ່ຄາດບໍ່ເຖິງໄດ້

- ຂໍ້ດີ : ລະບົບໃໝ່ສາມາດດໍາເນີນການໄດ້ທັນທີ, ສະຖານະການໄດ້ບັງຄັບໃຫ້ຜູ້ໃຊ້ໃຊ້ລະບົບ ໃໝ່ເຊິ່ງບໍ່ສາມາດກັບໄປໃຊ້ວຽກລະບົບເກົ່າໄດ້, ງ່າຍຕໍ່ການວາງແຜນ, ໃຊ້ເວລາໜ້ອຍ
- ຂໍ້ເສຍ : ອາດເກີດຂໍ້ຜິດພາດໄດ້ໃນລະຫວ່າງທີ່ໃຊ້ລະບົບໃໝ່, ເຖິງແມ່ນວ່າລະບົບໃໝ່ໃຊ້ວຽກ ໄດ້ຈຶ່ງແຕ່ບໍ່ໝາຍຄວາມວ່າຈະຄົບຖ້ວນ ແລະ ສົມບູນທຸກຢ່າງ, ມີຄວາມສ່ຽງສູງ

ການຕິດຕັ້ງລະບົບແບບຄູ່ຂະໜານ (Parallel Running)

ການຕິດຕັ້ງລະບົບແບບຄູ່ຂະໜານນັ້ນ ເປັນການປັບປ່ຽນທີ່ເພີ່ມພາລະວຽກໃຫ້ກັບພະນັກງານ ເປັນສອງເທົ່າ ເພາະວ່າໃນລະຫວ່າງທີ່ມີການໃຊ້ລະບົບໃໝ່ດ້ວຍການບັນທຶກ ຫຼື ປ້ອນຂໍ້ມູນໃຫ້ກັບລະບົບກໍຈະດໍາເນີນການເຮັດວຽກໃນລະບົບເດີມດ້ວຍ ຖ້າຫາກລະບົບໃໝ່ເກີດບັນຫາຂຶ້ນກໍຈະບໍ່ມີຜົນກະທົບໃດໆເພາະວ່າຍັງມີຂັ້ນຕອນຂອງການດໍາເນີນການໃນລະບົບເດີມຢູ່ການເຮັດວຽກແບບຄູ່ຂະໜານນີ້ຈະເຮັດວຽກຄຽງຄູ່ກັນໄປເລື້ອຍໆ ຈົນກວ່າຄົບກໍານົດເວລາທີ່ລະບົບໃໝ່ຈະສົມບູນສາມາດດໍາເນີນການ ໄດ້ຢ່າງບໍ່ມີບັນຫາການເຮັດວຽກແບບຄູ່ຂະໜານເປັນວຽກທີ່ໜັກຂອງພະນັກງານ ແລະ ຜູ້ວິເຄາະລະບົບທີ່ຕ້ອງເຂົ້າໄປດູແລແຕ່ກໍເປັນວິທີທີ່ນິຍົມ ແລະ ຖືເປັນການຕິດຕັ້ງລະບົບໃໝ່ທີ່ປອດໄພທີ່ສຸດໂດຍບໍ່ສ່ຽງ ຕໍ່ການຜິດພາດ.

- ຂໍ້ດີ: ມີຄວາມປອດໄພສູງ, ສາມາດປຽບທຽບຜົນຮັບຈາກລະບົບໃໝ່ ແລະ ລະບົບເດີມ ເຊິ່ງຈະເຫັນຜົນແຕກຕ່າງທີ່ຂ້ອນຂ້າງ ສັດເຈນ
- ຂໍ້ເສຍ: ໃຊ້ຕົ້ນທຶນສູງໃນສ່ວນຂອງການຕິດຕາມລະບົບທັງສອງລະບົບ, ເວລາທີ່ຕ້ອງສູນເສຍ ໄປກັບການເຮັດວຽກທັງສອງລະບົບ ແລະ ການປຽບທຽບ ທັງສອງລະບົບ, ຜູ້ໃຊ້ວຽກອາດຈະບໍ່ ຍອມຮັບໃນລະບົບໃໝ່ ເພາະຍັງເຮັດວຽກໃນລະບົບເດີມໄດ້ຢູ່, ການວາງແຜນ ແລະ ຄວບຄຸມມີຂັ້ນຕອນທີ່ຫຍຸ້ງຍາກ

ການຕິດຕັ້ງແບບເທື່ອລະເຟດ (Phase Changeover)

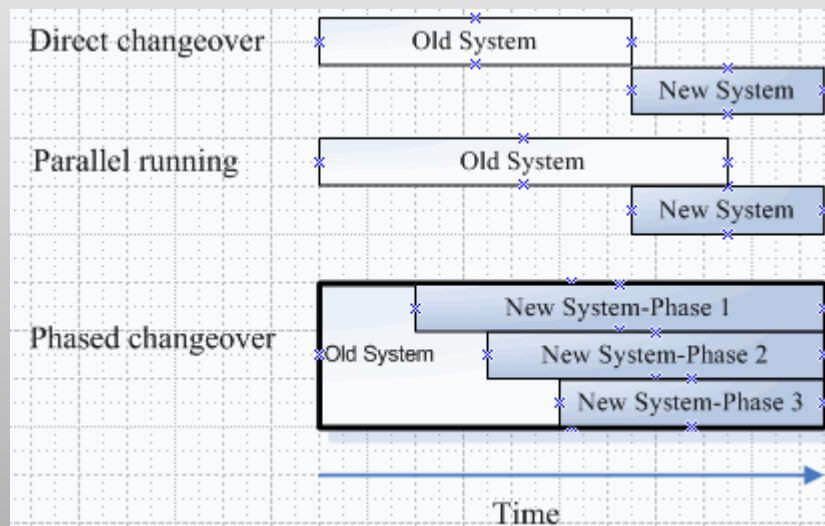
ການຕິດຕັ້ງແບບເທື່ອລະເຟດນັ້ນເປັນການຕິດຕັ້ງລະບົບເທື່ອລະລະບົບຍ່ອຍ (Sub System) ໂດຍຈະມີການກຳນົດການຕິດຕັ້ງໄປເທື່ອລະເຟດ ແລະ ຄ່ອຍໆດຳເນີນການໄປເລື້ອຍໆ ຈົນກະທັ້ງ

ຕິດຕັ້ງລະບົບຍ່ອຍໃນສ່ວນຕໍ່ໄປສຸດທ້າຍກໍຈະໄດ້ຄົບທຸກລະບົບ.

- ຂໍ້ດີ : ບໍ່ຈຳເປັນຕ້ອງຈ່າຍເງິນທັງໝົດສາມາດຈ່າຍເງິນຄ່າລະບົບເປັນສ່ວນໆ ຖ້າເກີດຂໍ້ຜິດ ພາດຈະບໍ່ກະທົບຕໍ່ລະບົບໂດຍລວມ. ແຕ່ຖ້າວ່າລະບົບຍ່ອຍນັ້ນຖືກຕ້ອງກໍສາມາດຕັດສິນໃຈ ດຳເນີນການຕໍ່ໃນເຟດຕໍ່ໄປ. ເໝາະກັບລະບົບວຽກທີ່ມີຂະໜາດໃຫຍ່
- ຂໍ້ເສຍ : ອາດໃຊ້ເວລາຫຼາຍເກີນໄປກັບວຽກໃນເຟດນັ້ນເຊິ່ງມີຜົນໃນການລໍຖ້າພັດທະນາໃນ ເຟດຕໍ່ໄປ

ການຕິດຕັ້ງແບບເປັນຕົວຢ່າງ (Pilot Project)

ການຕິດຕັ້ງແບບເປັນຕົວຢ່າງຈະຄ້າຍຄືກັບແບບເຟດແຕ່ເປັນການຕິດຕັ້ງລະບົບແບບໃຫ້ໃຊ້ວຽກສະເພາະສ່ວນໜຶ່ງກ່ອນເຊັ່ນວ່າ ການຕິດຕັ້ງເທື່ອລະພະແນກ ແລະ ຕິດຕາມຜົນງານ, ຖ້າປະສົບຜົນສໍາ ເລັດຈຶ່ງຂະຫຍາຍລະບົບໄປໃນສ່ວນອື່ນໆ



ການຈັດທຳເອກະສານ (Documentation)

ການຈັດທຳເອກະສານເປັນກິດຈະກຳໜຶ່ງທີ່ຕ້ອງເຮັດໃນໄລຍະການນຳໄປໃຊ້ , ແຕ່ບາງຄັ້ງເອກະສານເລົ່ານີ້ອາດຈະບໍ່ໄດ້ຮັບການສ້າງຂຶ້ນມາ ເພາະຄິດວ່າມັນບໍ່ສຳຄັນ ແຕ່ໃນຄວາມເປັນຈິງມັນມີຄວາມສຳຄັນຫຼາຍ ເນື່ອງຈາກເອກະສານເລົ່ານີ້ຈະເປັນຂໍ້ມູນສຳຄັນໃຫ້ຜູ້ໃຊ້ໄດ້ຮູ້ເຖິງວິທີການໃຊ້ວຽກ ແລະ ບຳລຸງຮັກສາ.

ເອກະສານໃນຮູບແບບຕ່າງໆ

- ເອກະສານອີເລັກໂທຣນິກ (Electronic Documents)

ເອກະສານຕ່າງໆທີ່ຖືກຈັດເກັບໃນຮູບແບບຂອງ ຟາຍ MS word, PDF

- ເອກະສານໄຮເປີລິງ (Hyperlinked Documents)

ເປັນເອກະສານທີ່ຖືກຈັດຮູບແບບຂຶ້ນມາໃຫ້ສາມາດເບິ່ງຜ່ານໂປຣແກຣມ

Browser ໂດຍຜູ້ອ່ານສາມາດກົດເບິ່ງຜ່ານ link ເພື່ອເຊື່ອມໂຍງໄປຫາຫົວເລື່ອງທີ່ຕ້ອງການໄດ້.

- ເອກະສານອອນໄລ (Online Documentation) ຖືກຈັດເກັບຢູ່ເທິງເວບໄຊຂອງຜູ້ຂາຍທີ່ຜູ້ໃຊ້ສາມາດເບິ່ງຜ່ານ Browser ຫຼື ດາວໂຫຼດເພື່ອໄປຕິດຕັ້ງໂປຣແກຣມເອກະສານຄູ່ມືມາເກັບໄວ້ໃນເຄື່ອງຂອງຜູ້ໃຊ້ເອງ

ເອກະສານໃນຮູບແບບຕ່າງໆ (ຕໍ່)

- ເອກະສານຖືກບັນທຶກເກັບລົງໃນສື່ (Embedded Documentation)

ເຊັ່ນ: CD ຫຼື DVD ໂດຍພາຍໃນຈະປະກອບດ້ວຍຄູ່ມື, ການຝຶກສອນ ແລະ ການນຳສະເໜີຜ່ານມັນຕີເມເດຍ

- ແບບຈຳລອງລະບົບອີເລັກໂທຣນິກ (Electronic System Models)

ເຊັ່ນ: ຂໍ້ຄວາມ ແລະ ຮູບພາບທີ່ຈັດເກັບຢູ່ໃນຮູບແບບຂອງຟາຍ GIF, JPEG ແລະ Visio

ຊະນິດຂອງເອກະສານ

ປະເພດຂອງເອກະສານແບ່ງອອກເປັນ 2 ຊະນິດຄື:

- ເອກະສານລະບົບ (System Documentation)
- ເອກະສານຜູ້ໃຊ້ (User Documentation)

ເອກະສານລະບົບ

ເປັນເອກະສານທີ່ເປັນຄູ່ມືທີ່ຊ່ວຍໃຫ້ຜູ້ດູແລລະບົບໄດ້ເຂົ້າໃຈກ່ຽວກັບຕົວໂປຣແກຣມນຳໃຊ້, ເຊິ່ງຄວບຄຸມກ່ຽວກັບສິ່ງຕໍ່ໄປນີ້:

- ແນະນຳທາງໃຫ້ການຈັດການກັບບັນຫາຕ່າງໆທີ່ອາດຈະເກີດຂຶ້ນພາຍຫຼັງຕິດຕັ້ງ
- ແນະນຳອັບເດດອຸປະກອນເພື່ອປະສິດທິພາບໃຫ້ກັບລະບົບ
- ວິທີກຳນົດສິດທິໃຫ້ກັບຜູ້ໃຊ້
- ແນະນຳກ່ຽວກັບການເພີ່ມ ຫຼື ເຄື່ອນຍ້າຍອຸປະກອນຈາກລະບົບ
- ວິທີສຳຮອງຂໍ້ມູນ ແລະ ກູ້ຄືນ
- ເອກະສານທີ່ກ່ຽວຂ້ອງກັບຊຸດຄຳສັ່ງ(source code) ເພື່ອໃຫ້ໂປຣແກຣມເນີນໄປປັບປຸງໃນອານາຄົດ

ເອກະສານຜູ້ໃຊ້

ເປັນເອກະສານທີ່ຊ່ວຍສະໜັບສະໜູນຜູ້ໃຊ້ໃຫ້ເຂົ້າໃຈເຖິງຂັ້ນຕອນກ່ຽວກັບການໃຊ້ວຽກຂອງລະບົບ ເຊິ່ງຄວບຄຸມກ່ຽວກັບສິ່ງຕໍ່ໄປນີ້:

- ບອກວິທີເປີດ/ປິດໂປຣແກຣມຢ່າງຖືກຕ້ອງ
- ບອກວິທີປ້ອນຂໍ້ມູນຢ່າງຖືກຕ້ອງ
- ບອກລຳດັບຂັ້ນຕອນໃນການປະມວນຜົນວ່າຈະຕ້ອງມີການປະຕິບັດແບບໃດຈົນກະທັ້ງໄດ້ຜົນຮັບຕາມຄວາມຕ້ອງການຂອງຜູ້ໃຊ້
- ບອກຂໍ້ຜິດພາດທີ່ອາດຈະເກີດຂຶ້ນ ແລະ ແນະນຳແນວທາງໃນການແກ້ໄຂ

ການບຳລຸງຮັກສາລະບົບ

ກິດຈະກຳໄລຍະນີ້ປະກອບດ້ວຍການ ການແກ້ໄຂໂປຣແກຣມ, ຂັ້ນຕອນ ຫຼື ເອກະສານຕ່າງໆເພື່ອໃຫ້ລະບົບທຳງານຕາມຄວາມຕ້ອງການຂອງຜູ້ໃຊ້ ແລະ ມີປະສິດທິພາບຫຼາຍຂຶ້ນ ເຊິ່ງກິດຈະກຳເລົ່ານີ້ປະກອບມີ:

- ການບຳລຸງຮັກສາດ້ວຍການແກ້ໄຂໃຫ້ຖືກຕ້ອງ
- ການບຳລຸງຮັກສາດ້ວຍການດັດແປງ
- ການບຳລຸງຮັກສາດ້ວຍການປັບປຸງ
- ການບຳລຸງຮັກສາດ້ວຍການປ້ອງກັນ

ການບຳລຸງຮັກສາດ້ວຍການແກ້ໄຂໃຫ້ຖືກຕ້ອງ (Corrective Maintenance)

ບາງເທື່ອຂໍ້ຜິດພາດທີ່ເກີດຂຶ້ນບໍ່ໄດ້ຖືກຄົ້ນພົບໃນລະຫວ່າງການທົດສອບ, ແຕ່ຄົ້ນພົບໃນລະຫວ່າງການໃຊ້ງານຈິງ ເຊິ່ງໂດຍປົກກະຕິຈະໄດ້ຮັບການແກ້ໄຂຢູ່ເລື້ອຍໆໃນຊ່ວງ 1-2 ປີທຳອິດ ຈົນຂໍ້ຜິດພາດໜ້ອຍລົງ ແລະ ໝົດໄປ.

ການບໍາລຸງຮັກສາດ້ວຍການດັດແປງ (Adaptive Maintenance)

ເປັນການບໍາລຸງຮັກສາໃຫ້ລະບົບສາມາດຮອງຮັບກັບສະພາບແວດລ້ອມໃໝ່ທີ່ມີການປ່ຽນແປງ ເຊັ່ນ: ກໍລະນີມີການປັບປ່ຽນອຸປະກອນ Hardware, ລະບົບປະຕິບັດການ ເຊິ່ງອາດຈະສົ່ງຜົນສະທ້ອນຕໍ່ລະບົບ ດັ່ງນັ້ນ ຈຶ່ງມີການແກ້ໄຂເພື່ອໃຫ້ລະບົບສາມາດດໍາເນີນງານໃນສະພາບແວດລ້ອມໃໝ່ໄດ້. ສໍາລັບການບໍາລຸງຮັກສາ ໃນໄລຍະທຳອິດຈະມີໜ້ອຍ, ແຕ່ເມື່ອມີການໃຊ້ງານດົນໄປ ແລະ ມີເຕັກໂນໂລຊີໃໝ່ໆເຂົ້າມາ ຈະເຮັດໃຫ້ການບໍາລຸງຮັກສາເພີ່ມຂຶ້ນ.

ການບໍາລຸງຮັກສາດ້ວຍການປັບປຸງ (Perfective Maintenance)

ເປັນການບໍາລຸງຮັກສາດ້ວຍການປັບປຸງຂະບວນການທີ່ມີຢູ່ໃຫ້ດີຂຶ້ນກວ່າເດີມ ເພື່ອໃຫ້ຜູ້ໃຊ້ສະດວກຂຶ້ນ ລວມທັງການເພີ່ມເຕີມ ຄຸນສົມບັດໃໝ່ໆ ເຂົ້າໄປໃນລະບົບ. ສໍາລັບກິດຈະກຳນີ້ຈະບໍ່ພົບໃນຊ່ວງທຳອິດຂອງການໃຊ້ງານ ແຕ່ຈະບົບຫຼາຍຂຶ້ນເລື້ອຍໆເມື່ອລະບົບຖືກໃຊ້ງານໄປໄລຍະໜຶ່ງ.

ການບໍາລຸງຮັກສາດ້ວຍການປ້ອງກັນ (Preventive Maintenance)

ເປັນການບໍາລຸງຮັກສາດ້ວຍການປ້ອງກັນບັນຫາທີ່ກໍາລັງຈະເກີດຂຶ້ນ
ໃນອານາຄົດ ເພື່ອຫຼີກລ້ຽງບັນຫາທີ່ຈະສົ່ງຜົນສະທ້ອນຕໍ່ລະບົບ
ເຊັ່ນ: ການອອກແບບລະບົບໃຫ້ສາມາດຮອງຮັບປະລິມານຂໍ້ມູນທີ່ມີ
ແນວໂນ້ມໃນອານາຄົດຈະເພີ່ມຂຶ້ນຢ່າງຫຼວງຫຼາຍ, ການອອກແບບ
ຄວາມປອດໄພເປັນຕົ້ນ. ໂດຍການບໍາລຸງຮັກສາດ້ວຍວິທີນີ້ຈະເກີດ
ຂຶ້ນໄດ້ທຸກເວລາຕະຫຼອດອາຍຸການໃຊ້ງານຂອງຊັອບແວ.

ນ້ຳໜັກຄວາມພະຍາມໃນການບຳລຸງຮັກສາ

ຊະນິດການບຳລຸງຮັກສາ	ລາຍລະອຽດ	ເປີເຊັນ
Corrective	ສ້ອມແຊມ ແລະ ແກ້ໄຂໂປຣແກຣມທີ່ຜິດພາດ	70%
Adaptive	ປັບປຸງລະບົບເພື່ອຮອງຮັບສະພາບແວດລ້ອມທີ່ປ່ຽນໄປ	10%
Perfective	ສ້າງລະບົບເພື່ອແກ້ໄຂບັນຫາໃໝ່ທີ່ເກີດຂຶ້ນ ຫຼື ປັບປຸງໃຫ້ດີຂຶ້ນກວ່າເກົ່າ	15%
Preventive	ປ້ອງກັນລະບົບຈາກບັນຫາອື່ນໆທີ່ອາດຈະເກີດຂຶ້ນໃນອານາຄົດ	5%

Thank you

Q and A