


252SE311: ວິສະວະກຳຊອບແວຣ໌ 2020-2021



| | |
|----------------|---|
| ອາຈານສອນ: | ບົວສິດ ໄຊຍະຈັກ |
| ວຸດທິການສຶກສາ: | ປະລິຍາໂທວິທະຍາສາດ (ວິທະຍາສາດຄອມພິວເຕີ) |
| ຕຳແໜ່ງບໍລິຫານ: | ຫົວໜ້າພະແນກການເງິນຂັ້ນສອງ |
| ໂທລະສັບມືຖື: | 22245134 |
| ອີເມລ: | bouasoth@yahoo.com |

ການປະເມີນຜົນ



| | |
|----------------------------|----|
| 1. ຄະແນນຂຶ້ນຫ້ອງ (Lec+Lab) | 20 |
| 2. ຄະແນນສ່ວນຮ່ວມໃນຫ້ອງ | 30 |
| 2.1 ກວດກາ(ກວດກາ+ວຽກບ້ານ) | 10 |
| 2.2 ບົດໂຄງການ | 20 |
| 3. ເສັງທ້າຍພາກ | 50 |

ຄວາມຮູ້ເບື້ອງຕົ້ນກ່ຽວກັບວິສະວະກຳຊອບແວຣ໌

ບົດທີ 1



ນຳສະເໜີກ່ຽວກັບວິສະວະກຳ ຊອບແວຣ໌

ເນື້ອໃນຫຍໍ້



- ◆ ຊອບແວຣ໌, ການປ່ຽນແປງ ແລະ ບັນຫາ
- ◆ ວິສະວະກຳຊອບແວຣ໌ ແລະ ຄວາມສຳຄັນ
- ◆ ອົງປະກອບຂອງວິສະວະກຳຊອບແວຣ໌
- ◆ ວິວັດທະນາການຂອງວິສາວະກຳຊອບແວຣ໌
- ◆ ຄຸນລັກສະນະຂອງຊອບແວຣ໌ທີ່ມີຄຸນນະພາບ
- ◆ ຫລັກການປະຕິບັດຂອງວິສະວະກຳຊອບແວຣ໌

ຊອບແວຣ໌, ການປ່ຽນແປງ ແລະ ບັນຫາ

- ◆ ນອກຈາກທີ່ຊອບແວຣ໌ໄດ້ມີບົດບາດໃນການອໍານວຍຄວາມສະດວກໃຫ້ແກ່ເຮັດວຽກຂອງມະນຸດແລ້ວ ມັນຍັງສາມສາດເປັນເຄື່ອງມືໃນການຊ່ວຍຜະລິດ ແລະ ຄວບຄຸມຊອບແວຣ໌ອື່ນໆໄດ້ອີກ
- ◆ ຊອບແວຣ໌ຖືກປ່ຽນແປງບົດບົດຜ່ານທາງຈຸດປະສົງການຜະລິດທີ່ແຕກຕ່າງກັນ (Program → Software → Application Software → Software Solution)
- ◆ ເມື່ອໄລຍະເວລາຜ່ານໄປຫລາຍປີ ຊອບແວຣ໌ທີ່ອົງກອນຕ່າງໆນໍາໄປໃຊ້ເລີ່ມມີການປ່ຽນແປງ, ບໍ່ເປັນລະບຽບ, ເຮັດວຽກຜິດພາດ, ແລະ ລ້າສະໄໝກາຍເປັນຊອບແວຣ໌ເກົ່າ ຊຶ່ງຈໍາເປັນຕ້ອງໄດ້ປັບປຸງ
- ◆ ບັນຫາ: ຊອບແວຣ໌ເກົ່າບໍ່ເໝາະສົມຕໍ່ການປັບປຸງ, ດັດແປງ, ການແກ້ໄຂ, ເຮັດໃຫ້ຕົ້ນທຶນສູງເກີນໄປ ເນື່ອງຈາກການອອກແບບທີ່ບໍ່ໄດ້ຄໍານຶງເຖິງການປ່ຽນແປງທີ່ເກີດຂຶ້ນໃນອານາຄົດທັງດ້ານຂະບວນການ ແລະ ຄວາມຕ້ອງການຂອງຜູ້ໃຊ້

ຊອບແວຣ໌, ການປ່ຽນແປງ ແລະ ບັນຫາ

◆ ປະເພດຂອງຊອບແວຣ໌

- System Software ແມ່ນບັນດາໂປຣແກຣມທີ່ໃຫ້ບໍລິການໂປຣແກຣມອື່ນ
- Application Software ແມ່ນໂປຣແກຣມແກ້ໄຂບັນຫາທາງທຸລະກິດ
- Scientific Software/Engineering ໃຊ້ສະເພາະວຽກວິທະຍາສາດແລະວກສ
- Embedded Software
- Product-Line Software
- We Application
- Artificial Intelligence

ວິສະວະກຳຊອບແວຣ໌ ແລະ ຄວາມສຳຄັນ

- ◆ ນິຍາມ 1: ວິສະວະກຳຊອບແວຣ໌ແມ່ນການໃຊ້ວິທີການທີ່ເປັນລະບົບ, ມີ ຫຼັກການ ແລະ ສາມາດວັດຜົນໃນທາງປະລິມານ ນຳມາປະຍຸກໃຊ້ໃນ ການພັດທະນາ, ປະຕິບັດການ ແລະ ບຳລຸງຮັກສາຊອບແວຣ໌ ຫຼື ສາມາດ ເວົ້າໄດ້ວ່າ ແມ່ນການສຶກສາວິທີການຜະລິດຊອບແວຣ໌
- ◆ ນິຍາມ 2: ແມ່ນການນຳໃຊ້ຫຼັກການວິຊາການດ້ານວິສະວະກຳມາໃຊ້ໃນ ຂະບວນການການຜະລິດຊອບແວຣ໌ ຕັ້ງແຕ່ຂັ້ນຕອນທຳອິດຈົນເຖິງຂັ້ນ ຕອນການບຳລຸງຮັກສາ ເພື່ອໃຫ້ຊອບແວຣ໌ທີ່ໄດ້ມີຄຸນນະພາບສູງສຸດພາຍ ໄຕ້ຂໍ້ຈຳກັດດ້ານເວລາ ແລະ ຕົ້ນທຶນ

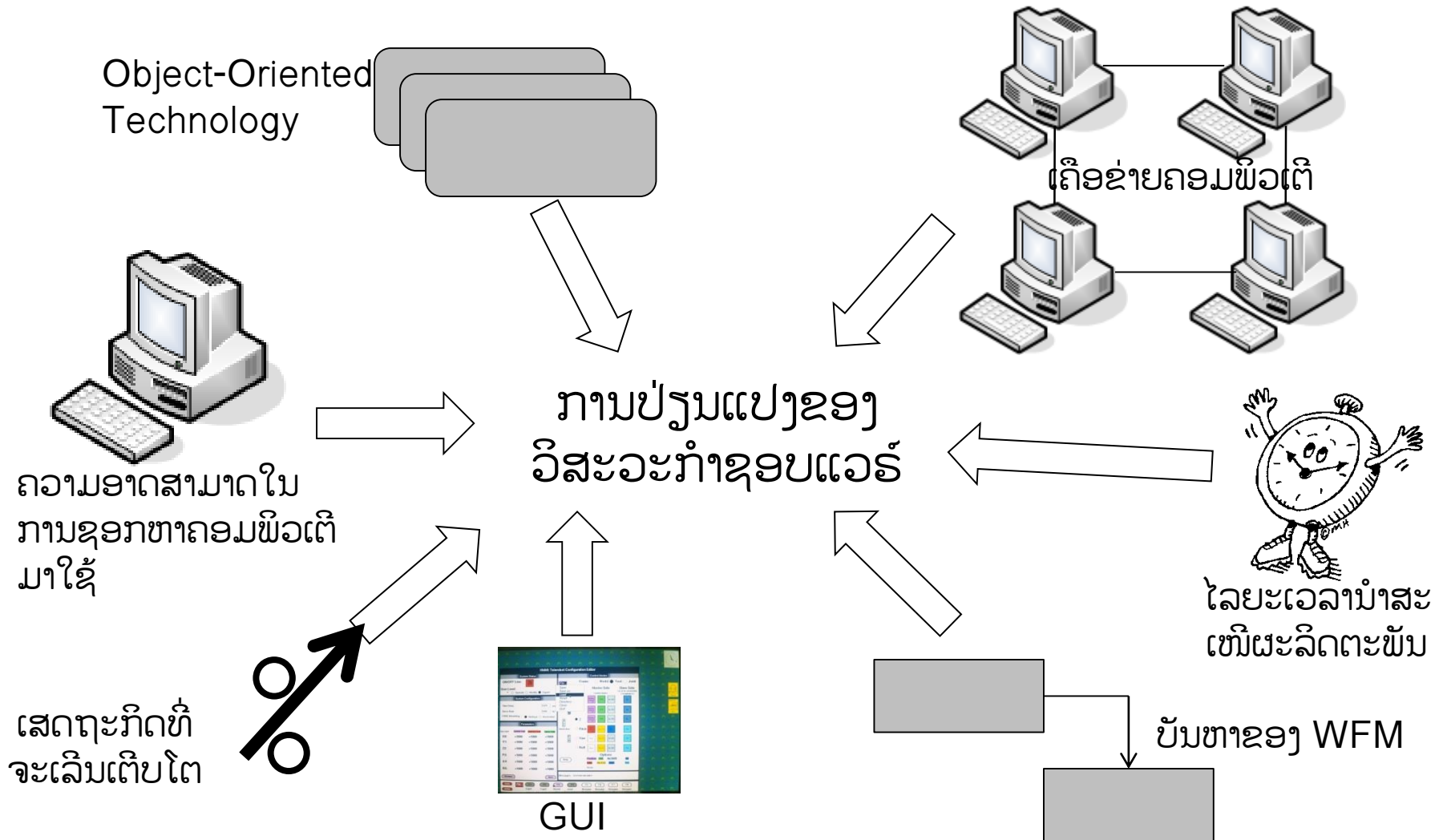
ວິສະວະກຳຊອບແວຣ໌ ແລະ ຄວາມສຳຄັນ

◆ ຄວາມສຳຄັນ

- ◆ ວິສະວະກຳຊອບແວຣ໌ໄດ້ປັບປຸງຂະບວນການຜະລິດຊອບແວຣ໌ໃໝ່ໃຫ້ມີປະສິດທິພາບກ່ວາເກົ່າ ໂດຍການນຳໃຊ້ ທິດສະດີ, ເຕັກນິກ, ເຄື່ອງມື ແລະ ວິທີການເຮັດວຽກໃນແຕ່ລະຂັ້ນຕອນ ທີ່ຈະເຮັດໃຫ້ສາມາດຜະລິດຊອບແວຣ໌ໄດ້ຢ່າງວ່ອງໄວ, ມີຄຸນນະພາບ ແລະ ແກ້ໄຂດັດແປງໄດ້ງ່າຍ ເພື່ອບໍ່ໃຫ້ຊອບແວຣ໌ກາຍເປັນສາເຫດຂອງຄວາມຜິດພາດ ທີ່ອາດກໍ່ໃຫ້ເກີດຄວາມເສຍຫາຍທາງດ້ານຊີວິດ ແລະ ສັບສິນເໝືອນໃນອາດິດ

ວິສະວະກຳຊອບແວຣ໌ ແລະ ຄວາມສຳຄັນ

ປັດໃຈການປ່ຽນແປງທີ່ເຮັດໃຫ້ວິສະວະກຳຊອບແວຣ໌ມີຄວາມສຳຄັນ
ຫລາຍຂຶ້ນ

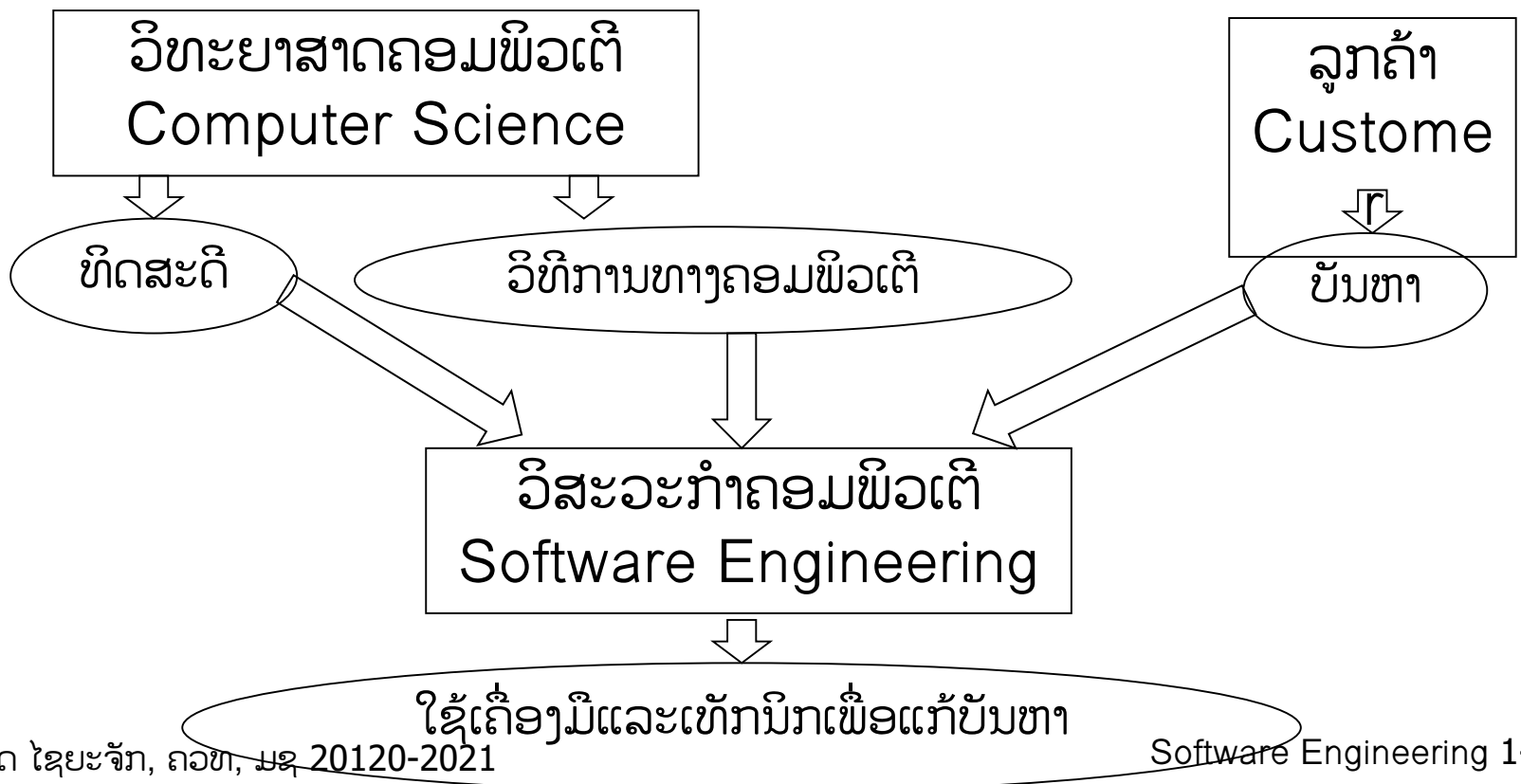


ວິສະວະກຳຊອບແວຣ໌ ແລະ ຄວາມສຳຄັນ

- ◆ ຄວາມແຕກຕ່າງລະຫວ່າງວິສະວະກຳຊອບແວຣ໌ ແລະ ວິທະຍາສາດຄອມພິວເຕີ
 - ◆ ວິທະຍາສາດຄອມພິວເຕີເປັນວິຊາທີ່ວ່າດ້ວຍທິດສະດີ ແລະ ວິທີການທາງດ້ານຄອມພິວເຕີ ແລະ ລະບົບຊອບແວຣ໌ທີ່ເລິກເຊິ່ງ
 - ◆ ວິສະວະກຳຊອບແວຣ໌ເປັນວິຊາທີ່ວ່າດ້ວຍບັນຫາໃນທາງປະຕິ ບັດຕົວຈິງຂອງການຜະລິດຊອບແວຣ໌ ບໍ່ວ່າຈະເປັນການບໍລິຫານຈັດການຊັບພະຍາກອນ, ການອອກແບບລະບົບຫຼືຊອບ ແວຣ໌, ການເລືອກເທັກໂນໂລຢີ, ຜົນກະທົບຂອງສະພາບແວດລ້ອມທີ່ມີຕໍ່ລະບົບ, ການປະເມີນດ້ານຕ່າງໆ, ການທົດສອບ, ການປະກັນຄຸນນະພາບ ແລະ ການບຳລຸງຮັກສາລະບົບ

ວິສະວະກຳຊອບແວຣ໌ ແລະ ຄວາມສຳຄັນ

- ◆ ຄວາມແຕກຕ່າງລະຫວ່າງວິສະວະກຳຊອບແວຣ໌ ແລະ ວິທະຍາສາດຄອມພິວເຕີ



ວິສະວະກຳຊອບແວຣ໌ ແລະ ຄວາມສຳຄັນ



- ◆ ຄວາມແຕກຕ່າງລະຫວ່າງວິສະວະກຳຊອບແວຣ໌ ແລະ ວິສະວະກຳລະບົບ
 - ◆ ວິສະວະກຳລະບົບກ່ຽວຂ້ອງກັບທຸກໆດ້ານຂອງການພັດທະນາ ແລະ ການປ່ຽນແປງຂອງລະບົບທີ່ມີຄວາມຊັບຊ້ອນ ໂດຍມີຊອບແວຣ໌ເປັນແກນຫຼັກໃນການເຮັດວຽກຂອງລະບົບ
 - ◆ ວິສະວະກຳລະບົບກ່ຽວຂ້ອງກັບການພັດທະນາຮາດແວຣ໌, ນະໂຍບາຍ, ອອກແບບ, ພັດທະນາຂັ້ນຕອນ ແລະ ລະບົບໄປພ້ອມໆກັບການເຮັດວິສະວະກຳຊອບແວຣ໌

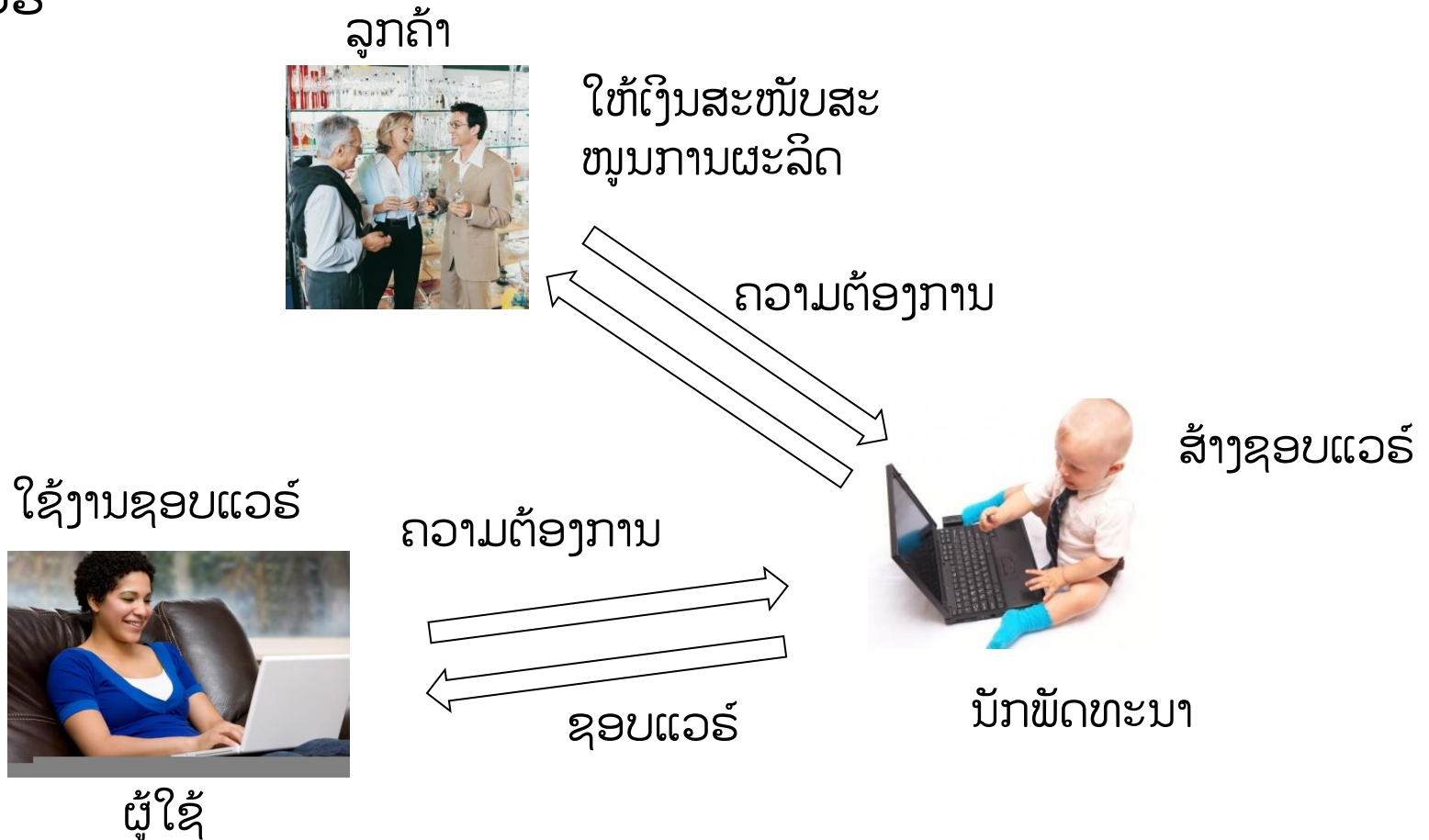
ວິສະວະກຳຊອບແວ ແລະ ຄວາມສຳຄັນ



- ◆ ບຸກຄົນທີ່ມີສ່ວນກ່ຽວຂ້ອງກັບວິສະວະກຳຊອບແວ
 - ◆ ລູກຄ້າ (Customer)
 - ◆ ນັກພັດທະນາ (Developer-Software Engineer)
 - ◆ ຜູ້ໃຊ້ (User)

ວິສະວະກຳຊອບແວຣ໌ ແລະ ຄວາມສຳຄັນ

- ◆ ຄວາມສຳພັນລະຫວ່າງກຸ່ມບຸກຄົນທີ່ມີສ່ວນກ່ຽວຂ້ອງກັບວິສະວະກຳຊອບແວຣ໌



ວິສະວະກຳຊອບແວ ແລະ ຄວາມສຳຄັນ

◆ ຄວາມຮູ້, ຄວາມສາມາດ ແລະ ທັກສະດ້ານຕ່າງໆ ສຳລັບວິສະວະກອນຊອບແວ

◆ ຄວາມຮູ້ດ້ານການຜະລິດຊອບແວ

◆ ຄວາມຮູ້ດ້ານການບໍລິຫານໂຄງການ

◆ ຄວາມຮູ້ດ້ານການຈັດການ

◆ ຄວາມຮູ້ດ້ານທຸລະກິດ

◆ ຄວາມຮູ້ດ້ານປະຊາສຳພັນ

◆ ຄວາມໜ້າເຊື່ອຖື

◆ ຄວາມຮູ້ສຶກໄວ

◆ ຄວາມເປັນຜູ້ນຳ

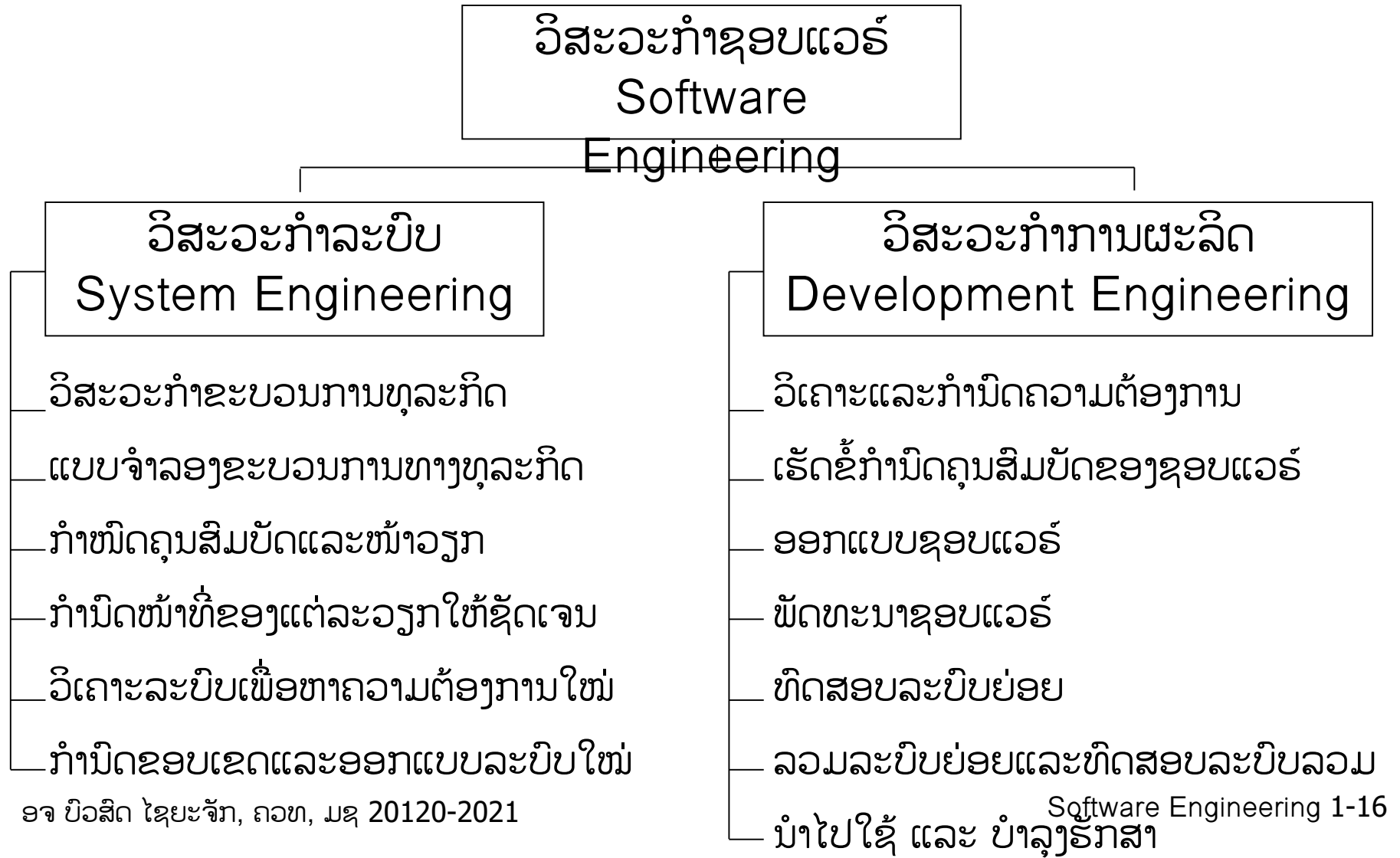
◆ ຄວາມອົດທົນຕໍ່ພາວະຄວາມກົດດັນ

◆ ຄວາມຢືດຢຸນສູງ

◆ ຄວາມຮັບຜິດຊອບສູງ

◆ ຄວາມຍຸດຕິທຳ

ອົງປະກອບຂອງວິສະວະກຳຊອບແວ



ອົງປະກອບຂອງວິສະວະກຳຊອບແວຣ໌

- ▶ ໃນສ່ວນຂອງວິສະວະກຳລະບົບມີກິດຈະກຳຕ່າງໆດັ່ງນີ້
 - ▶ ກຳນົດຈຸດປະສົງຂອງລະບົບ
 - ▶ ກຳນົດຂອບເຂດຂອງລະບົບ
 - ▶ ແບ່ງລະບົບອອກເປັນພາກສ່ວນຍ່ອຍ ຕາມໜ້າທີ່ການທຳງານ ແລະ ພຶດຕິກຳຂອງລະບົບ
 - ▶ ພິຈາລະນາຄວາມສຳພັນຂອງສ່ວນປະກອບຕ່າງໆ
 - ▶ ກຳນົດຄວາມສຳພັນຂອງປັດໃຈທີ່ສົ່ງເຂົ້າ, ປະມວນຜົນ ແລະ ຜົນໄດ້ຮັບ
 - ▶ ພິຈາລະນາປັດໃຈທີ່ມີສ່ວນກ່ຽວຂ້ອງໃນລະບົບ(Hardware, Software, Database, Other software)
 - ▶ ກຳນົດຄວາມຕ້ອງການຂອງ Operation ແລະ Function ຂອງທັງໝົດລະບົບ
 - ▶ ສ້າງແບບຈຳລອງລະບົບ ເພື່ອໃຊ້ວິເຄາະ ແລະ ພັດທະນາໃຫ້ຊອດຄ້ອງກັບແບບຈຳລອງຊອບແວຣ໌ທີ່ສ້າງຂຶ້ນ
 - ▶ ນຳສະເໜີ ແລະ ແລກປ່ຽນຂໍ້ຄິດເຫັນກັບຜູ້ທີ່ກ່ຽວຂ້ອງກັບລະບົບ (ຜູ້ໃຊ້, ເຈົ້າຂອງ, ກ່ຽວຂ້ອງກັບຜົນປະໂຫຍດ)

ອົງປະກອບຂອງວິສະວະກຳຊອບແວຣ໌

- ▶ ໃນສ່ວນຂອງວິສະວະກຳການຜະລິດມີກິດຈະກຳຕ່າງໆດັ່ງນີ້
 - ▶ ກຳນົດຄວາມຕ້ອງການ ແລະ ເຮັດຂໍ້ກຳນົດຄຸນສົມບັດ
 - ▶ ອອກແບບວິທີທາງແກ້ໄຂບັນຫາໃຫ້ຖືກຕ້ອງຊອດຄ່ອງກັບຄວາມຕ້ອງການ
 - ▶ ພິຈາລະນາສະຖາປັດຕະຍະກຳໃຫ້ຊອດຄ່ອງກັບວິທີທາງແກ້ໄຂບັນຫາ
 - ▶ ວາງແຜນໂຄງການຜະລິດຊອບແວຣ໌
 - ▶ ທົດສອບຊອບແວຣ໌ໃນແຕ່ລະສ່ວນຍ່ອຍ
 - ▶ ລວມສ່ວນຍ່ອຍຕ່າງໆເຂົ້າເປັນລະບົບດຽວກັນ
 - ▶ ທົດສອບລະບົບລວມ ພ້ອມກວດສອບຄວາມຖືກຕ້ອງ ແລະ ຄວາມຊອດຄ່ອງກັບຄວາມຕ້ອງການທີ່ໄດ້ກຳນົດໄວ້
 - ▶ ພິຈາລະນາວິທີການທີ່ຈະນຳໄປໃຊ້
 - ▶ ນຳໄປໃຊ້
 - ▶ ປັບປຸງຂັ້ນຕອນການຈັດການ
 - ▶ ບຳລຸງຮັກສາ ແລະ ຕິດຕັ້ງຊອບແວຣ໌

ວິວັດທະນາການຂອງວິສາວະກຳຊອບແວ

◆ 1945-1965

- ◆ ເປັນຈຸດເລີ່ມຕົ້ນຂອງວິສາວະກຳຊອບແວຊຶ່ງກ່ຽວຂ້ອງກັບວິສະວະກຳຄອມພິວເຕີ, ເອເລັກໂທຣນິກ ແລະ ສະຖາປະນິກ

◆ 1966-1985

- ◆ ເປັນຈຸດວິກິດຂອງວິສະວະກຳຊອບແວ ເຊັ່ນ: OS/360 ໃຊ້ເວລາພັດທະນາຍາວນານໂພດ, ຊອບແວລະບົບຄວາມປອດໄພຂອງບ່ອນເກັບຈະລວດນຳວິຖີມີຂໍ້ບົກພ່ອງ, ຊອບແວລະບົບຄວບຄຸມການແຜ່ລັງສີຂອງເຄື່ອງຮັກສາຄົນເຈັບດ້ວຍລັງສີວິທະຍາບໍ່ເຮັດວຽກໃນເວລາປະຕິບັດງານ

◆ 1986-ປະຈຸບັນ

- ◆ ເປັນຍຸກຂອງການແກ້ໄຂບັນຫາວິກິດທາງດ້ານຊອບແວ ໂດຍມີປັດໃຈທີ່ເປັນແຮງຂັບເຄື່ອນດັ່ງນີ້: Tools, Techniques, Interdiscipline, Professionalism

ຄຸນລັກສະນະຂອງຊອບແວທີ່ມີຄຸນນະພາບ



- ◆ ຄວາມຖືກຕ້ອງ (Correctness)
- ◆ ສະດວກສະບາຍຕໍ່ຜູ້ໃຊ້ (User Friendly)
- ◆ ບໍາລຸງຮັກສາງ່າຍ (Maintainability)
- ◆ ຄວາມໜ້າເຊື່ອຖື (Reliability)
- ◆ ມີປະສິດທິພາບ (Efficiency)
- ◆ ເຮັດໃຫ້ສະດວກ ແລະ ງ່າຍຕໍ່ການໃຊ້ງານ (Usability)
- ◆ ສາມາດນຳກັບມາໃຊ້ໃໝ່ໄດ້ (Reusability)
- ◆ ມີຄວາມແຂງແຮງທົນທານ (Robustness)
- ◆ ນຳໄປຕິດຕັ້ງກັບທຸກລະບົບໄດ້ (Portability)
- ◆ ມີຄວາມປອດໄພ (Security)

ຫລັກການປະຕິບັດຂອງວິສະວະກຳຊອບແວຣ໌

◆ ຫລັກການແບບໂຄງສ້າງ (Structure Approach)

- ເປັນວິທີການແບ່ງລະບົບ ແລະ ຄວາມຕ້ອງການອອກເປັນລະບົບຍ່ອຍ ຕາມລັກສະນະຂອງໜ້າທີ່ແຕ່ລະວຽກ. ແຕ່ລະລະບົບຍ່ອຍສາມາດແບ່ງອອກເປັນລະບົບຍ່ອຍລົງໄປເລື້ອຍໆຖ້າເຫັນວ່າຍັງມີຄວາມຊັບຊ້ອນຢູ່
- ລັກສະນະຂອງລະບົບໂຄງສ້າງຈຶ່ງເປັນແບບລຳດັບຊັ້ນ
- ຫລັກການທີ່ນິຍົມໃຊ້ໃນການວິເຄາະແລະ ອອກແບບລະບົບກໍຄື Structure System Analysis and Design (SSAD) ທີ່ຄິດຄົ້ນໂດຍ Yourdan & Demarco
- ການວິເຄາະ ແລະ ອອກແບບ ຂໍ້ມູນ, ພຶດຕິກຳຂອງລະບົບແມ່ນແຍກອອກຈາກກັນ ເຮັດໃຫ້ເສຍເວລາ, ຕົ້ນທຶນ ແລະ ມີຄວາມສ່ຽງສູງ

ຫລັກການປະຕິບັດຂອງວິສະວະກຳຊອບແວຣ໌

◆ ຫລັກການແບບວັດຖຸ (Object-Oriented Approach)

- ເປັນວິທີການໃໝ່ ທີ່ຄິດຄົ້ນໂດຍ Grady Booch, James Rumbaugh ແລະ Ivar jacobson ດ້ວຍວິທີການວິເຄາະແລະອອກແບບໃນທາງວັດຖຸ (Object-Oriented System Analysis and Design) ເປັນການວິເຄາະໂດຍການເບິ່ງທຸກຢ່າງເປັນວັດຖຸ (Object) ຊຶ່ງພາຍໃນວັດຖຸນັ້ນ ຈະມີທັງຂໍ້ມູນ ແລະ ພຶດຕິກຳຢູ່ນຳກັນ
- ເຮັດໃຫ້ການວິເຄາະ ແລະ ອອກແບບລະບົບໄວຂຶ້ນ
- ມີຄວາມນິຍົມສູງໃນປະຈຸບັນ ແລະ ມີແນວໂນ້ມເພີ່ມຂຶ້ນໃນອານາຄົດ