

Dasturiy ta'minotni ishlab chiqish bosqichlari

Alisher Shakirovich Ismailov
Abdulhamid Akbarov
Gulnoza To'xtasinjon qizi Qodirova
Maftuna Yigitaliyeva
Toshkent moliya instituti

Annotatsiya: Kichik dasturiy masala yoki muammolarni yechish ancha sodda. Lekin katta dasturlar yaratilganda ularni talablarini o'rganish, masalaga yechim topish, algoritmlar ishlab chiqish katta jamoa va ko'p vaqt talab etadi. Bugungi kunda katta dasturlar yaratishda dasturlarni o'z muddatida topshirish uchun tizim yaratilgan. Ushbu tizim SDLC (Software development life cycle) Dastur yaratish hayot sikli deb ataladi.

Kalit so'zlar: dasturiy ta'minot, dasturiy ta'minot ishlab chiqish bosqichlari, loyiha, SDLC

Stages of software development

Alisher Shakirovich Ismailov
Abdulhamid Akbarov
Gulnoza Tokhtasinjan kizi Kadirova
Maftuna Yigitaliyeva
Tashkent Financial Institute

Abstract: Solving small software issues or problems is much simpler. But when large programs are created, it takes a large team and a lot of time to study their requirements, find a solution to the problem, and develop algorithms. Today, when creating large programs, a system is created to deliver programs on time. This system is called SDLC (Software development life cycle).

Keywords: software, software development stages, project, SDLC

1. SDLC (Software development life cycle)

SDLC - dasturiy ta'minotni yaratish uchun tizimli jarayon bo'lib, u yaratilgan dasturiy ta'minotning sifati va to'g'riligini ta'minlaydi. SDLCning maqsadi mijozlar talablariga javob beradigan yuqori sifatli dasturiy ta'minotni ishlab chiqarishga qaratilgan. Tizimni ishlab chiqish oldindan belgilangan vaqt va belgilangan narx doirasida tugallanishi kerak. SDLC batafsil rejadan iborat hisoblanadi, u aniq dasturiy

ta'minotni rejalashtirish, yaratish va saqlashni tushuntiradi. SDLC hayotiy siklining har bir bosqichi o'z jarayoni va keyingi bosqichga o'tadigan natijalarga ega bo'ladi.

2. SDLC ahamiyati

SDLC dasturiy ta'minot tizimini ishlab chiqish uchun muhim bo'lgan asosiy sabablar quyidagilardan iborat.

- SDLC loyihani rejalashtirish va baholash uchun asosni beradi
- Standart faoliyat va natijalar to'plami uchun asosni taqdim etadi
- SDLC loyihani kuzatish va nazorat qilish mexanizmi hisoblanadi
- Loyihani rejalashtirish jarayonining barcha manfaatdor tomonlari uchun ko'rinishini oshiradi
- Dasturni ishlab chiqish tezligini oshiradi
- Mijoz va dasturchi aloqalarini muntazamligini ta'minlaydi
- Loyihani boshqarish rejasini qo'shimcha harajatlarini kamaytirishga yordam beradi

3. SDLC bosqichlari

1-bosqich: Texnik talablarni o'rganib chiqish va tahlil qilish

2-bosqich: Texnik-iqtisodiy asoslarni ishlab chiqish (Loyihaning (dastur) umumiy byudjeti aniqlanadi)

3-bosqich: Dizayn (algoritm yaratish)

4-bosqich: Kodlash

5-bosqich: Sinov (har bir funksiya to'g'ri ishlayotgani tekshiriladi)

6-bosqich: O'rnatish (dasturni mijozning qurilmasiga yoki hostingga o'rnatish jarayoni)

7-bosqich: Texnik xizmat ko'rsatish (dasturga kelajakdagi o'zgartirishlar kiritish xizmatlari)

1-bosqich: Dastur talablarini yig'ish va tahlil qilish

Talablarni aniqlash - SDLC jarayonining birinchi bosqichi hisoblanadi. U barcha manfaatdor tomonlar va soha mutaxassislarining ma'lumotlari bilan olib boriladi. Ushbu bosqichda dasturning har bir talabi (requirement) ni o'rganib chiqish va rejalashtirish amalga oshiriladi. Ushbu bosqich butun loyihaning ko'lami va loyihani ishga tushirganda kutilayotgan muammolar, imkoniyatlar va ko'rsatmalar haqida aniqroq tasavvur beradi. Talablarni yig'ish bosqichida batafsil va aniq talablarni olish uchun jamoa kerak bo'ladi. Bu kompaniyalarga yaratilayotgan yangi dasturiy loyihani ishini tugatish uchun kerakli bo'lgan vaqt muddatini tashkil etishga yordam beradi.

2-bosqich: Texnik-iqtisodiy asoslash

Talablarni tahlil qilish bosqichi tugallangandan so'ng, keyingi SDLC bosqichi dasturiy ta'minot ehtiyojlarini aniqlash va hujjatlashtirish hisoblanadi. Ushbu jarayon "SRS" hujjati sifatida ham tanilgan, "Dasturiy ta'minot talablari spetsifikatsiyasi" hujjati yordamida amalga oshiriladi. U loyihaning hayot sikli davomida ishlab

chiqilishi kerak bo'lgan barcha narsalarni o'z ichiga oladi. Ushbu bosqichning asosiy foydasi bu ishlab chiqilishi rejalashtirilgan dasturning amalga oshira olamizmi degan savolga javob berishi hisoblanadi. Bu fizibiliti deb ham ataladi.

Fizibiliti tekshiruvlarining asosan besh turi mavjud:

1. Iqtisodiy: Loyihani byudjet doirasida yakunlay olamizmi yoki yo'qmi?
2. Yuridik: Biz ushbu loyihani kiber qonun va boshqa me'yoriy-huquqiy ba'za/muvofiqliklar sifatida ko'rib chiqa olamizmi?
3. Operatsion fizibiliti: Biz mijoz kutgan operatsiyalarni yarata olamizmi?
4. Texnik: joriy qurilma tizimi dasturiy ta'minotni qo'llab-quvvatlay oladimi yoki yo'qligini tekshirish kerak
5. Vaqt muddati: Loyihani belgilangan vaqt muddatida bajarish mumkin yoki yo'qligini aniqlash.

3-bosqich: Dizayn

Ushbu uchinchi bosqichda tizim va dasturiy ta'minotni loyihalash hujjatlari talabning spetsifikatsiya hujjatiga muvofiq tayyorlanadi. Bu umumiy tizim arxitekturasi aniqlashga yordam beradi. Dizayn bosqichi modelning keyingi bosqichi uchun kirish bo'lib xizmat qiladi. Dizayn bosqichida algoritmlar ham yaratiladi. Ushbu bosqichda ishlab chiqilgan ikki turdagi dizayn hujjatlari mavjud bo'lishi kerak:

Yuqori darajadagi dizayn (HLD)

- Har bir modulning qisqacha tavsifi va nomi
- Har bir modulning funksionalligi haqida qisqacha ma'lumot
- Interfeys aloqasi va modullar orasidagi bog'liqliklar
- Ma'lumotlar ba'zasi jadvallari va ularning asosiy elementlari
- To'liq arxitektura diagrammalari va texnologiya tafsilotlari

Quyi darajadagi dizayn (LLD)

- Modullarning funksional mantiqi
- Turi va hajmini o'z ichiga olgan ma'lumotlar ba'zasi jadvallari
- Interfeysning to'liq tafsilotlari
- Xato xabarlar ro'yxati
- Har bir modul uchun to'liq kirish va chiqish ma'lumotlari

4-bosqich: Kod yozish

Tizimni loyihalash bosqichi tugagach, keyingi bosqich kod yozish hisoblanadi. Ushbu bosqichda ishlab chiquvchilar tanlangan dasturlash tilidan foydalangan holda kod yozish orqali butun tizimni (dasturni) qurishni boshlaydilar. Kodlash bosqichida vazifalar birliklarga yoki modullarga bo'linadi va turli ishlab chiquvchilarga beriladi. Bu dasturiy ta'minotni ishlab chiqishning hayot sikli jarayonining eng uzun bosqichi hisoblanadi. Ushbu bosqichda dasturchi 1-2-3-bosqichlarda aniqlab tayyorlangan ko'rsatmalariga amal qilishi kerak. Shuningdek, ular kodni yaratish va amalga oshirish

uchun kompilyator, tarjimon, tuzatuvchi kabi dasturlash vositalaridan foydalanishlari ham kerak bo'ladi.

5-bosqich: Sinov

Dastur kodlari yozilib tayyor bo'lgandan so'ng, u sinov bosqichida tekshiriladi. Odatda sinov alohida firma/kompaniya xizmati yordamida amalga oshiriladi. Sinov o'tkazish guruhi yangi yaratilgan butun dasturning funkcionalligini sinab ko'rishni boshlaydi. Bu butun dastur mijoz talabiga muvofiq ishlashini tekshirish uchun amalga oshiriladi. Ushbu bosqichda QA va sinov guruhi ishlab chiquvchilarga (dasturchilarga) xabar beradigan ba'zi xatolar/nuqsonlarni topishi mumkin. Ishlab chiqish guruhi xatoni tuzatadi va qayta sinovdan o'tkazish uchun QAga yuboradi. Bu jarayon yaratilgan dastur xatosiz, barqaror va ushbu tizimning biznes ehtiyojlariga muvofiq ishlamaguncha davom etadi.

6-bosqich: O'rnatish/joylashtirish

Dasturni sinovdan o'tkazish bosqichi tugagach va tizimda hech qanday xato qolmaganidan so'ng, yakuniy joylashtirish jarayoni boshlanadi. Loyiha menejeri tomonidan berilgan fikr-mulohazalar asosida yakuniy dastur versiyasi joylashtirish bilan bog'liq muammolar tekshiriladi. Agar desktop dastur yaratilgan bo'lsa mijozning qurilmasiga o'rnatib beriladi. Agar yaratilgan dastur internetda ishlaydigan bo'lsa server (hosting)ga joylashtiriladi va mijozga topshiriladi. Ushbu bosqichda ishlab chiquvchi dasturchilarning shartnomaga doir ishlari yakunlangani belgilanadi. Va agar mijoz ishlab chiquvchiga yangi o'zgartirishlar uchun xizmat qilishni taklif qilsa keyingi bosqich ham ishlab chiquvchiga tegishli hisoblanadi.

7-bosqich: Texnik xizmat ko'rsatish

Yaratilgan dastur o'rnatilgach va mijozlar ishlab chiqilgan tizimdan foydalanishni boshlagandan so'ng, quyidagi 3 ta harakat sodir bo'ladi.

- Xatolarni tuzatish - xatoliklar haqida umuman tekshirilmagan ayrim senariylar tufayli xabar qilinadi

- Yangilash - dasturni yangi versiyalariga yangilash

- Yaxshilash - mavjud dasturiy ta'minotga ba'zi yangi xususiyatlarni qo'shish

Ushbu SDLC bosqichining asosiy yo'nalishi ehtiyojlarni qondirishda davom etishini va dastur birinchi bosqichda aytib o'tilgan spetsifikatsiyaga muvofiq ishlashini ta'minlashdan iborat hisoblanadi.

Xulosa

Dasturiy ta'minotni ishlab chiqish hayot sikli (SDLC) dasturiy ta'minotni yaratish uchun tizimli jarayon bo'lib, u yaratilgan dasturiy ta'minotning sifati va to'g'riligini ta'minlaydi. Dasturiy ta'minot injiniringidagi SDLC standart faoliyat va natijalar to'plami uchun asos yaratadi. Yetti xil SDLC bosqichlari: 1) Talablarni yig'ish va tahlil qilish 2) Texnik-iqtisodiy asoslash 3) Dizayn 4) Kodlash 5) Sinov 6) O'rnatish / Joylashtirish va 7) Texnik xizmat ko'rsatishdan iboratdir.

Foydalanilgan adabiyotlar

1. Ismailov, A., Jalil, M. A., Abdullah, Z., & Abd Rahim, N. H. (2016, August). A comparative study of stemming algorithms for use with the Uzbek language. In 2016 3rd International conference on computer and information sciences (ICCOINS) (pp. 7-12). IEEE.
2. Jalil, M. M., Ismailov, A., Abd Rahim, N. H., & Abdullah, Z. (2017). The Development of the Uzbek Stemming Algorithm. *Advanced Science Letters*, 23(5), 4171-4174.
3. Abdurakhmonova, N., Alisher, I., & Sayfulleyeva, R. (2022, September). MorphUz: Morphological Analyzer for the Uzbek Language. In 2022 7th International Conference on Computer Science and Engineering (UBMK) (pp. 61-66). IEEE.
4. Ismailov, A. S., & Jo'rayev, Z. B. Study of arduino microcontroller board.
5. Ismailov, A. S., Alijanov, D. D., Jo'rayev, Z. B., & Kurbanov, M. U. Research on renewable energy sources in Uzbekistan.
6. Ismailov, A. S., Shamsiyeva, G., Abdurakhmonova, N., & Navoi, A. Statistical machine translation proposal for Uzbek to English.
7. Abdurakhmonova, N. Z., Ismailov, A. S., & Mengliev, D. (2022, November). Developing NLP Tool for Linguistic Analysis of Turkic Languages. In 2022 IEEE International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON) (pp. 1790-1793). IEEE.
8. Abdurakhmonovaa, N., Alisher, I., & Toirovaa, G. (2022, September). Applying Web Crawler Technologies for Compiling Parallel Corpora as one Stage of Natural Language Processing. In 2022 7th International Conference on Computer Science and Engineering (UBMK) (pp. 73-75). IEEE.
9. Abdurakhmonova, N., Tuliyeu, U., Ismailov, A., & Abduvahobo, G. (2022). UZBEK ELECTRONIC CORPUS AS A TOOL FOR LINGUISTIC ANALYSIS. In *Компьютерная обработка тюркских языков. TURKLANG 2022* (pp. 231-240).
10. Abdurakhmonova, N., Shakirovich, I. A., & O'G'Li, K. N. S. (2022). Morphological analyzer (morfoAnalyse) Python package for Turkic language. *Science and Education*, 3(9), 146-156.
11. Ismailov, A. S., & Abdurakhmonova, N. The development of Alisher stemmer for Uzbek Language.
12. Ismailov, A. S., Abdurakhmonova, N., & Tojiboyev, G. O. (2021). Wireless traffic light controller. *Science and Education*, 2(12), 249-253.
13. Shakirovich, I. A. OZBEK STEMASINI YARATISH.