**BTEC Assignment Brief**

|  |  |
| --- | --- |
| **Malaka** | Pearson BTEC Higher Nationals in Digital Technologies |
| **Fan raqami va nomi** | Unit 4: Programming |
| **O'rganish maqsadlari** (faqat NQF uchun) | Ushbu bo‘limning oxiriga kelib, talabalar quyidagilarni bajara oladilar:  LO1 Amalni bajarish uchun asosiy algoritmlarni belgilash va ilovani dasturlash jarayonini tasvirlash.  LO2 Protsedurali, obyektga yo‘naltirilgan va hodisaga asoslangan dasturlashning xususiyatlarini tushuntirish.  LO3 Integratsiyalashgan dasturlash muhitidan (IDE) foydalanib, asosiy algoritmlarni kodda amalga oshirish.  LO4 Xatoliklarni tuzatish jarayonini aniqlash va kodlash standartining ahamiyatini tushuntirish. |
| **Topshiriq sarlavhasi** | Bog'cha uchun Ovqatlar va Ombor Hisoboti Dasturi |
| **Baholovchi** | Shukurov Sanjar |
| **Berilgan sanasi** |  |
| **Belgilangan muddatda topshiring** |  |
|  | |
|  | |
| **Kasbiy ssenariy yoki kontekst** | **Loyiha tavsifi:** Sizga bog'cha uchun to'liq dasturiy yechim yaratish topshirilgan. Bu dastur orqali oshxona xodimlari mahsulotlarni hisobga olish, ovqatlar tarkibini boshqarish, hisobotlarni ko'rish imkoniyatiga ega bo'ladi. Bu tizim real hayotdagi ehtiyojlarni aks ettirishi va quyidagi funksiyalarni o'z ichiga olishi kerak:  **Funktsional talablar:**   1. **Mahsulotlar boshqaruvi (Masalan, go'sht, kartoshka, tuz):**    * Mahsulot nomi va grammda umumiy miqdori kiritiladi.    * Mahsulot miqdori yangilanadi yoki o'chiriladi.    * Yetkazilgan sanasi kiritiladi. 2. **Ovqatlar boshqaruvi:**    * Ovqat nomi va har bir ingredient uchun kerakli miqdor belgilanadi.    * Ovqatlar va retseptlar yangilanadi yoki o'chiriladi. 3. **Ovqat berish tizimi:**    * "Ovqat berildi" tugmasi bosilganda ingredientlar ombordan kerakli miqdorda ayriladi.    * Agar ingredient yetarli bo'lmasa, xatolik xabari chiqadi.    * Ovqat berilgan sana/vaqt va kim tomonidan berilgani qayd qilinadi. 4. **Porsiya hisoblash:**    * Har bir ovqatdan mavjud mahsulotlar asosida nechta porsiya tayyorlash mumkinligi ko'rsatiladi.    * Bu hisob har safar ingredientlar o'zgarishi bilan dinamik ravishda yangilanadi. 5. **Vizualizatsiya va hisobotlar:**    * Ingredientlar iste'moli va kelib tushgan sanalari bo'yicha grafiklar.    * Oylik hisobotlarda:      + Tayyorlangan porsiyalar soni.      + Tayyorlanishi mumkin bo'lgan porsiyalar soni.      + Farq foizi (%). Agar 10–15% dan oshsa, ehtimoliy suiste'mol haqida signal bo'lishi kerak. 6. **Foydalanuvchilarni kuzatish:**    * Kim qaysi ovqatni berganligi, sana va vaqt bilan qayd qilinadi.    * Jadvalda barcha ovqat berish holatlari ko'rsatiladi. 7. **Rolga asoslangan kirish:**    * **Admin:** Hammasiga kirish, sozlamalar, hisobotlar.    * **Oshpaz:** Faqat ovqat berish imkoniyati.    * **Menejer:** Omborni yangilash, tahlillarni ko'rish. 8. **Ogohlantirishlar:**    * Ingredient miqdori belgilangan minimumdan kamaysa, xabar chiqadi.    * Agar oylik farq 15% dan oshsa, bu haqida ogohlantirish beriladi. 9. **Fon vazifalari (Celery):**    * Oylik hisobotlarni avtomatik yaratish.    * Porsiya taxminlarini qayta hisoblash. 10. **Real-vaqt yangilanishlar (WebSocket):**  * Ombor holatini jonli ko'rsatish. * Ovqat berish vaqtida ingredient tugayotgan bo'lsa, real-vaqt ogohlantirish.   **Texnologiyalar bo‘yicha tavsiyalar:**   * Backend: Django (DRF) yoki FastAPI * Frontend: React, Vue.js yoki HTML/CSS + Chart.js * Ma'lumotlar bazasi: PostgreSQL (yaxshiroq), SQLite (soddalik uchun) * Vazifa navbati: Celery + Redis * Grafiklar: Chart.js yoki Recharts * Real-vaqt: Django Channels yoki FastAPI WebSocket   **Tavsiyalangan bosqichlar:**   1. Ingredient va ovqat CRUD (formalar va DB modellari) 2. Ovqat berish funksiyasi va loglash 3. Porsiya hisoblash algoritmini qo‘shish 4. Vizualizatsiya va oylik hisobotlar 5. (Qo‘shimcha) Rollar, ogohlantirishlar, WebSocket integratsiyasi   **Topshiriladigan fayllar:**   * To‘liq manba kodi (backend + frontend) * Ma’lumotlar bazasi sxemasi * Misol ma’lumotlar (fixtures) * Muhim sahifalarning skrinshotlari * README fayli (o‘rnatish bo‘yicha yo‘riqnoma)   **Baholash mezonlari:**   * Funktsional to‘g‘rilik (ingredientlar ayrilishi, porsiya hisoblash, grafiklar) * Kodni tashkil qilish va aniqlik * Tegishli texnologiyalar va kutubxonalarni tanlash * Ijodkorlik va mustahkamlik (bonus funksiyalar) * Ma’lumotlar yaxlitligi va xavfsizlik (rolga asoslangan kirish, validatsiya) |
|  | |

**O‘rganish natijalari va baholash mezonlari**

|  |  |  |
| --- | --- | --- |
| **Pass** | **Merit** | **Distinction** |
| **LO1 Operatsiyani bajarish uchun asosiy algoritmlarni aniqlang va dasturlashtirish jarayonini bayon qiling** | | **D1** Mos tilda algoritm amalga oshirilishini va yozma algoritm bilan kod varianti o‘rtasidagi munosabatni  baholang. |
| **P1** Algoritmning ta’rifini bering va ilova yaratish  Jarayonini batafsil tushuntiring. | **M1** Kod yozishdan ijro etishgacha bo‘lgan bosqichlarni aniqlang. |
| **LO2** Protsessual, obyekt-yo‘naltirilgan va hodisalarga asoslangan dasturlash xususiyatlarini tushuntiring | | **D2** Protsessual, obyektga yo‘naltirilgan va hodisaga asoslangan paradigmalarni amalga oshiradigan dastur manba kodini, uning tuzilishi va xususiyatlari nuqtai nazaridan tanqidiy baholang. |
| **P2** Jarayonga asoslangan, obyektga yo‘naltirilgan va hodisaga asoslangan paradigmalarning mohiyati, ularning o‘ziga xos xususiyatlari hamda o‘zaro aloqadorligi haqida tushuntirish bering. | **M2** Dasturning berilgan manba kodida qo‘llaniladigan protsessual, obyektga yo‘naltirilgan va hodisaga asoslangan paradigmalarni  taqqoslab, ularning farqlarini ko‘rsating |
| **LO3** IDE yordamida asosiy algoritmlarni kodda amalga oshirish | | **D3** Ilovalarni ishlab chiqishda IDE dan foydalanish va foydalanmaslik o‘rtasidagi farqni tahlil qiling va baholang. |
| **P3** IDE yordamida algoritm bajaradigan dastur yozing. | **M3** Dasturni ishlab chiqish jarayonini boshqarish uchun IDE’dan foydalaning. |
| **LO4** Debaglash jarayonini belgilang va kodlash standartining muhimligini tushuntirib bering | | **D4** Jamoada va alohida shaxs uchun kodlash standartining nima uchun zarurligini tanqidiy baholang. |
| **P4** IDEda mavjud bo‘lgan nosozliklarni tuzatish jarayoni va nosozliklarni aniqlash vositalarini tushuntirib bering | **M4** Xavfsiz va mustahkam dasturlarni yaratishda nosozliklarni bartaraf etish jarayonining qanday yordam bera olishini baholang. |
| **P5** Kodingizda qo‘llagan kodlash standartini batafsil ko‘rsating. |

|  |  |
| --- | --- |
| **Ushbu topshiriqni bajarishingizda yordam beradigan ma’lumot manbalari** | **Topshirish talablari:**  • P1, P2, P4, P5, M1, M2, M3, M4, D1, D2, D3, D4 uchun yozma hisobotlar.  • M1 uchun blok-sxemalar yoki diagrammalar.  • Chakana savdo buyurtma tizimi uchun manba kod fayllari.  • Loyiha davomida yuzaga kelgan qiyinchiliklar, o‘rganilgan saboqlar va mulohazalarni qayd etuvchi rivojlanish kundaligi.  **Kitoblar:**  Dennis, A. va Haley, W. (2009) Tizimlarni tahlil qilish va loyihalash. John Wiley & Sons Ltd.  Ferguson, J. (2014) BDD amalda: Butun dasturiy ta’minot hayot sikli uchun xulq-atvorga asoslangan ishlab chiqish. Manning.  Lejk, M. va Deeks, D. (2002) Tizimlarni tahlil qilish usullariga kirish. 2-nashr. Addison-Wesley.  Murch, R. (2012) Dasturiy ta’minot ishlab chiqarish hayot sikli: To‘liq qo‘llanma. Kindle. |