**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** | Chakana savdo buyurtma berish tizimi |
| **Baholovchi** | Shukurov Sanjar |
| **Berilgan sanasi** |  |
| **Belgilangan muddatda topshiring** |  |
|  | |
|  | |
| **Kasbiy ssenariy yoki kontekst** | Ssenariy:  NewEra Cash & Carry - bu Toshkentning joylashgan, 5000 nafar mijoz bazasiga ega bo‘lgan xayoliy chakana oziq-ovqat do‘koni. U mijozlarga non mahsulotlari, ichimliklar, dori-darmonlar, go‘sht, dengiz mahsulotlari va boshqa ko‘plab oziq-ovqat mahsulotlarini sotadi. Hozirgi vaqtda do‘kon kundalik operatsiyalarni qo‘lda boshqarish tizimidan foydalanmoqda. Bu esa ma’lumotlarning noaniqligi, samarasizligi, ishonchsizligi va takrorlanishiga olib keladi. Biznes rahbariyati do‘konning kundalik operatsiyalarini boshqarish uchun kompyuterga asoslangan ma’lumotlar bazasi tizimi orqali jarayonlarni takomillashtirmoqchi. NewEra Cash & Carry chakana buyurtma tizimini avtomatlashtirish bo‘yicha yordam berish uchun sizga murojaat qilishdi. Asosiy biznes jarayon ssenariysi quyidagicha: Yuqoridagi tizim orqali oziq-ovqat mahsulotlariga buyurtma bermoqchi bo‘lgan mijozlarning buyurtma jarayonini boshqarish imkoniyatiga ega bo‘lishi kerak. Tizimga ma’lumotlarni qo‘shish, o‘zgartirish va o‘chirish huquqiga ega bo‘lgan administrator mavjud bo‘lishi lozim.  API Talablari  Administrator imkoniyatlari:   * Autentifikatsiya: Xavfsiz admin kirish nuqtasi. * Mahsulotlar CRUD: Mahsulotlarni yaratish, o'qish, yangilash va o'chirish uchun nuqtalar:   + POST /api/products - Yangi mahsulot qo'shish   + GET /api/products/{id} - Mahsulot tafsilotlarini olish   + PUT /api/products/{id} - Mahsulotni yangilash   + DELETE /api/products/{id} - Mahsulotni o'chirish * Buyurtma boshqaruvi:   + GET /api/orders - Barcha buyurtmalarni ro'yxatini olish   + GET /api/orders/{id} - Aniq buyurtma tafsilotlarini ko'rish   Mijozlar uchun imkoniyatlari:   * Autentifikatsiya: Xavfsiz mijoz kirish nuqtasi. * Mahsulotlar ro'yxati:   + GET /api/products - Barcha mahsulotlar ro'yxatini olish * Buyurtma tizimi:   + POST /api/orders - Yangi buyurtma yaratish (mahsulot IDlari va miqdori bilan)   + GET /api/orders/{customerId} - Aniq mijozning buyurtmalarini olish   + GET /api/orders/{orderId}/status - Buyurtma holatini tekshirish   Vazifa Tafsilotlari  1-qism: API Dizayni va Amalga Oshirilishi   1. API Hujjatlari:    * API nuqtalari, parametrlari, javob kodlari va misollarni qayd etish uchun Swaggerdan foydalaning. 2. Amalga Oshirish:    * Yangi .NET Core Web API loyihasini o'rnatish.    * JWT (JSON Web Tokens) orqali admin va mijoz rollari uchun autentifikatsiya amalga oshirish.    * Ma'lumotlar bazasi sxemasini dizayn qilish (ORM uchun Entity Framework Core dan foydalanishingiz mumkin):      + Products, Orders, OrderDetails (buyurtmalar uchun), Users jadvallari.    * Har bir API nuqtasi uchun kontrolerlar yozish:      + Samaradorlik uchun asinxron metodlardan foydalaning.      + Xatolarni boshqarish va loglashni amalga oshirish. 3. Sinovlar:    * xUnit yoki NUnit kabi sinov dasturlari bilan xizmat va kontroler qatlamlari uchun birlik sinovlarini yozish.   2-qism: Optimizatsiya va Xavfsizlik   1. Ish hajmini optimizatsiya qilish:    * Ma'lumotlar bazasi so'rovlarini tahlil qilish va optimizatsiya qilish, masalan, LINQ dan samarali foydalanish.    * Kerakli joylarda kesh qilish (masalan, mahsulotlar ro'yxati uchun). 2. Xavfsizlikni kuchaytirish:    * Barcha nuqtalarda kirish ma'lumotlarini tekshirish.    * Barcha aloqalarda HTTPS dan foydalanish.    * Foydalanuvchi ma'lumotlari va tokenlarni xavfsiz boshqarish. 3. API versiyalash:    * Kelajakdagi yangilanishlar uchun API versiyalashni o'rnatish.   3-qism: Ishga Tushirishga Tayyorlik   1. Ishchi muhit uchun konfiguratsiya:    * Bog'liqlik satrlari kabi konfiguratsiya sozlamalari uchun atrof-muhit o'zgaruvchilaridan foydalanish.    * Turli muhitlar uchun osongina sozlanadigan log qilishni o'rnatish. 2. Dockerization:    * Dasturni osongina ishga tushirish uchun DockerFile yaratish.   Topshiriqlar:   * Kod bazasi: To'liq ishlaydigan .NET Core Web API loyihasi. * Hujjatlar: /swagger orqali kirish mumkin bo'lgan Swagger hujjatlari. * Sinovlar: Asosiy komponentlarning ishlashini ko'rsatadigan birlik sinovlari to'plami. * Hisobot: Quyidagilar haqida gapirish:   + Qanday qilib qarorlarni qabul qilingan (nega ma'lum kutubxonalar yoki usullar ishlatilgan).   + Amalga oshirilgan xavfsizlik choralari.   + Samaradorlik haqidagi mulohazalar va optimizatsiyalar.   + Kelajakda API qanday kengaytirilishi yoki miqyoslanishi mumkinligi. |
|  | |

**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. |
| --- | --- |