|  |  |
| --- | --- |
| Слайд - Назва | Текст |
| 1 - Вступ | Всім привіт. Мене звати Тарас. Ми група 273.Net. І Сьогодні ми б хотіли представити нашу розробку, над якою ми працювали протягом останніх 3-ох місяців. |
| 2-Команда | Для початку я б хотів познайомити Вас з нашою командою. Зі мною ви вже знайомі, мене звати Тарас. Це Іра, Наталя, Анастасія, Іван, Андрій та Василь. Також тут присутній наш експерт Ростислав, та наша викладачка Леся. |
| 3-Проблема | Напевно більшість з тих хто тут присутній, стикався з проблемою при Вступі у ВНЗ. Яку університет, галузь чи спеціальність обрати? |
| 4-Вирішення | Вирішенням цієї проблеми є наш сайт який дозволяє переглядати усі університети України з їх спеціальностями, проходити профорієнтаційними тестами.  Наш сайт орієнтований на учнів старших класів та їх батьків. |
| 5-Ролі | В нашій системі передбачається 3 ролі – відвідувач, Зареєстрований користувач та адмін. На даний момент роль адміністратора не реалізована |
| 6-Фунціональність Відвідувача | Користувач може переглядати список усіх університетів, спеціальності в цих університетах, переглядати Найкращі університети, проходити проф-орієнтаційні тести, переглядати спеціальності за допомогою пошуку по галузях, та по предметах. Також йому надається можливість зареєструватись та аутентфікуватись у системі. |
| 7 – Функціональність зареєстрованого користувача | Що стосується функціональності зареєстрованого користувача то він має ту ж саму функціональність що й відвідувач, а також зареєстрований користувач може додавати спеціальності в список улюблених, йому автоматично зберігаються результати проф-орієнтаційних тестів. Цю інформацію він може переглянути в персональному кабінеті. Також він може змінювати свій пароль. |
| 8 – Аналоги | Під час розробки системи ми також аналізували ринок схожих систем |
| 9 – Vstup.info | Першим з яких є Vstup.info – на цьому сайті можливо переглянути універсистети України, а також спеціальності в них. Великим недоліком цієї системи незручний інтерфейс. |
| 10– Abit-poisk | Наступним аналогом є abit-poisk, його перевагою над Vstup.info є можливість пошуку спеціальностей за певними фільтрами |
| 11 – Технології | А зараз я вам розкажу про технології які ми використовували.  Середовищем розробки ми використовували VS 2017 Community, .Net Core Framework, a в якості ORM – EF Core.  БД ми використовували MS SQL SERVER, наш сайт задеплоєний на MS Azure, а також ми використовували MS IIS  На фронтенді було використано бібліотеку React, HTML, CSS та Bootstrap  Для Source Control був обраний Git, з клієнтом SmartGit та в якості репозиторію ми використовували GitHub.  Для комунікації в команді ми використовували Slack  Для тестування api-запитів ми використовували Postman і в якості профайлеру – Prefix. Методологією розробки ПЗ – SCRUM |
| 12 - Репозиторій | Перейдемо до нашого проекту. На картинці зверху зображено структуру нашого репозиторію. В папці documentation зберігається документація по нашому проекту. В education зберігаються навчальні матеріали які ми використовували. В папці prototypes містяться невеличкі прототипні проекти. А папка SRC це папка нашого проекту. На картинці нижче зображено вміст цієї папки що відповідає нашим проектам в цьому рішенні. |
| 13 – DB, WEB | В проекті DB можна виділити такі папки як Model в яких міститяться моделі нашої БД, Migration – міграції БД, SQLDataAccess – Методи доступу до БД.  В проекті WEB виділю папку ClientApp – тут містяться компоненти React а також CSS файли, Controller - тут зберігаються класи контроллерів, а також хотів би виокремити файл Startup.cs де зберігаються налаштування проекту, в тому числі і DI про які розкаже вже моя колега Анастасія. |
| 45-Профорієнтаційні тести | Для проходження профорієнтаційних тестів, користувачу необхідно на навігаційній панелі, натиснути «Профорієнтаційні Тести», і тоді він перейде на сторінку з списком доступних тестів. |
| 48- Профорієнтаційні тести React | Зверху можна побачити api-запит який дістає дані про профорієнтаційні тести.  Знизу виділено як ці дані заповнюються в таблиці на |
| 49 - Test detail component | Після вибору певного тесту користувач переходить на сторінку з детальною інформацією про тест, де він може з ним ознайомитись, і при бажані розпочати |
| 50 - Test detail component React | Зверху ми дістаємо параметр для отримання даних про цей тест х props, формуємо з нього шлях для api-запиту. А нижче відбувається отримування даних про тест |
| 63 – БД для отримання спеціальностей | Важливою компонентою системи є можливість переглядати спеціальності та інформацію про них, а також список предметів ЗНО які необхідно здати для вступу на цю спеціальність. Ця частинка використовується практично в кожній функціональності системи, тому для цього було створено частину БД яка зображена на проекторі. Таблиця Specialties містить інформацію про всі спеціальності в Україні, таблиця Subjects містить інформацію про предмети які здають для вступу в ВНЗ. А проміжна таблиця Specialty-Subject містить інформацію про те яка спеціальність вимагає які предмети для вступу. |
| 79 – обрати спеціальність за предметами DAL | Для отримання списку предметів за обраними предметами на DAL спочатку обираються id-шки спеціальностей які містять ті предмети які обрав користувач.  Після того, вибираються інформація про спеціальності. В кінці можна побачити Skip.Take - що використовується для виведення обмеженої кількості предметів на сторінку користувача. |
| 95-Персональний кабінет | Коли користувач ввійшов в систему йому стає доступна можливість перегляду персонального кабінету. Для цього користувачеві необхідно натиснути кнопку «Персональний кабінет» на навігаційній пенелі |
| 97- персональний кабінет API | Для того що б дістати персональну інформацію про користувача ви беремо його ідентифікатор з токена і тоді вже витягуємо його персональну інформацію |
| 98 – персональний кабінет реакт | Ось так виглядає скелет цієї компонента на реакті, тут ми формуємо блок з його персональною інформацією та блок з його вподобаннями та пройденими профорієнтаційними тестами |
| 99-зміна паролю | Також користувачеві в персональному кабінеті доступна можливість зміни паролю |
| 100 | Для зміни паролю я використав метод з UserMeneger Якщо користувач ввів правильний поточний пароль, в такому випадку зміна паролю пройде успішно, в протилежному йому прийде повідомлення з тим що операція провалилась. |
| 101 | Ось так це відбувається на Reacti. Паролі хешуються і передаються Post методом на бекенд. Перед тим варто зазначити що ці паролі також валідуються. |
| 102 | В персональному кабінеті також може переглянути результати тестів які він проходив з пропозиціями до них. Для цього йому необхідно натиснути та тест який він хоче обрати в списку результатів тестів |
| 107 | В персональному кабінеті користувач може переглянути спеціальності які він уподобав |
| 112 | На Реакті було використано компоненті List Specialties для виводу елементів списку а також Pagination |
| 113 | Ось так виглядає сторінка обраних спеціальностей користувача |
| 114 - Кінець | На цьому наша доповідь завершена. Ми дякуємо за увагу. І можливо у Вас є Ще питання? |