

Документация на проект WebStol

Съдържание

Документация на проект WebStol.....	1
1. ТЕМАWebStol.....	2
2. АВТОРИ.....	2
3. РЪКОВОДИТЕЛ	2
4. РЕЗЮМЕ	3
4.1 Цели	3
4.2 Основни етапи в реализирането на проекта.....	3
- student.....	4
- administrator	4
- JavaScript (основен език в приложенията).....	5
- Babel.....	5
4.3 Ниво на сложност на проекта – основни проблеми при реализацията на поставените цели.....	6
4.4 Логическо и функционално описание.....	7
4.5 Реализация	7
4.6. Описание на приложението	9
Инсталиране на настолното приложение.....	9
Използване на настолното приложение	11
Меню прозореца.	12
Списъкът с ученици	13
Месечен отчет:	15
Инсталиране на мобилното приложение за Андроид	17
Работа с мобилното приложение.....	18
Ученически интерфейс	19
Учителски интерфейс	20
4.7. Заключение	22

1. ТЕМА WebStol

Идеята на проекта е система, чрез която всеки ученик, учител и класен ръководител ще може да заяви да ползва услугите за столово хранене. Ще може да получава бързи справки за бъдещо меню, за предишни ползвания на услуги и сумата, която дължи до момента.

2. АВТОРИ

Калин Димитров Паралеев

0445037243

Хаджи Генчо №13 гр. Копривщица

0887580371

kalin.paraleev.2023@soulkaravelov.com

СУ „Любен Каравелов“ гр. Копривщица

12 клас

Георги Ивайлов Иванов

0441076724

Нешо Шабанов №15 гр. Копривщица

0896217431

georgi.ivanov.2023@soulkaravelov.com

СУ „Любен Каравелов“ гр. Копривщица

12 клас

3. РЪКОВОДИТЕЛ

Стоян Динчев Павлов

0889016689

s.pavlov@sulkaravelov.com

Учител професионална подготовка

4. РЕЗЮМЕ

4.1 Цели

Преди появата на този продукт, всяка сутрин работник от ДСП Копривщица, с тефтер и химикал, в ръка обикаляше коридорите на училището, записваше желаещите ученици и учители да ползват услугите им, като едновременно с това им съобщаваше дължимата към момента сума и ежедневните цени на храненето. Главната идея беше да направим нещо несъществуващо до момента, но да е полезно и да спестим труд на определен човек или на определена група от хора.

Целите на готовия продукт са:

- Да се оптимизира работата на ДСП гр. Копривщица в частта на столовотохранене на учениците на СУ “Любен Каравелов“, гр. Копривщица;
- Да се улесни достъпа на учениците на СУ “Любен Каравелов“, гр. Копривщица до столовото хранене и да се улесни заявяването на обяд;
- Да се събират на 100% дължимите такси от учениците на училището към ДСП гр. Копривщица;
- Учителите от начален курс да могат лесно да заявят за учениците желание за хранене;
- Лесно и бързо получаване на различни отчети;
- Достъп до седмично меню и цени по всяко време, една седмица предварително;

Предназначението на продукта е да се заявява ползване на услугите за хранене в стола от една страна и генерирането на отчети за дължими суми и брой ползвания от друга страна.

4.2 Основни етапи в реализирането на проекта

1. Запознаване с необходимостта за системата;
2. Уточняване на необходимите нива на достъп;
3. Избор на технологии;
4. Изготвяне;
5. Тестване/Обучаване на персонала;
6. Въвеждане в експлоатация.

1. Запознаване с необходимостта за система

На среща с директора и работниците в Дирекция Социално подпомагане гр. Копривщица установихме необходимостта от подобно разпределено приложение.

Разгледахме и проучихме внимателно работата им и уточнихме какво трябва да се разработи, и какви трябва да бъдат входните данни и евентуалните необходими отчети. Обмислихме, кои са местата, където трябва да бъде отделено повече време за работа.

2. Уточняване на необходимите нива на достъп

Нужните нива на достъп са 3:

- **student**
- **teacher**
- **administrator**

На всеки потребител ще бъде предоставен акаунт в зависимост от различните нива на достъп. Акаунтите използват имейл и парола, чрез които потребителите ще могат да влизат в акаунтите си и да използват системата. Акаунтите имат 3 нива на достъп с различни права и функции. Учениците имат акаунт с ниво на достъп „student“, класните ръководители (1-4 клас) – „teacher“, а главния работник в столовата – „administrator“.

student – Това ниво на достъп има стриктно ограничени права, чрез които може да променя и преглежда единствено информация, която е свързана с акаунта използван от потребителя. Това включва:

- Заявка, чрез която показва дали през утрешния ден ще използва услугите на столаили не;
- Преглед на меню за следващия ден;
- Преглед на цена на храната за следващия ден;
- Преглед на брой посещения - единствено за себе си;
- Преглед на цената, която дължи до момента – единствено за себе си;

teacher – Нивото на достъп „teacher“ има права, чрез които може да променя и преглежда единствено информация, която е свързана с акаунта използван от потребителя, както и да променя и преглежда информация, свързана с всички акаунти на ученици от своя клас. Това включва:

- Всичко от ниво на достъп „student“;
- Заявка, чрез която показва дали даден ученик ще използва услугите на столаили няма да използва утрешния ден ;
- Преглед на брой посещения на даден ученик;
- Преглед на цената, която дължи до момента даден ученик;

По този начин децата, които нямат възможност да използват приложението няма да бъдат оцетени и с помощта на класния им ръководител ще могат да ползват всички функции на системата.

administrator – Това е нивото на достъп с най-много права. Може да промени информация свързана с другите 2 нива на достъп. Променя менюто, както и цената за всяко хранене. Задава дати за месечен отчет и дати за почивни дни, в които столовата не работи.

3. Избор на технологии

Системата има 2 интерфейса. Мобилно приложение (Android/iOS) за ученици и класни ръководители и десктоп приложение (Windows/macOS) за администратора.

Използвани технологии за мобилното приложение:

- *React Native* (Front-end Framework) + *Expo* (bundle of tools created around React Native)
- *Firebase* (Database)

Използвани технологии за десктоп приложение:

- *React JS* (Front-end Framework) + *Electron JS* (Software Development Framework за направата на native applications на Windows, macOS и Linux чрез веб- технологии като HTML, CSS и JavaScript)
- *Firebase* (Database)

Други технологии:

- **JavaScript (основен език в приложенията)**
- **Node JS**
- **Babel**
- **Dotenv**

4. Изготвяне

Георги Ивайлов Иванов

Главна дейност - Front-end / UI/UX (User Interface/User Experience)

5. Тестване/Обучаване на персонала

За да сме сигурни, че приложенията са готови за ползване от крайните потребители, те са тествани със симулирани ситуации, с цел откриване на бъгове, грешки в кода и несъответствие на информация. Въпреки направените от разработчиците тестове, сценариите при интерактивността на потребителите са много, а грешките - неизбежни.

Поради тази причина имаме за цел да разширим тестването, като предварително пуснем системата в работа под формата на тест. Ще бъдат инсталирани определен брой приложения на устройствата на ученици, класни ръководители, както и администраторски интерфейс на един от работниците в столовата. За да използват приложенията, ще бъде нужно персонала да се обучи, а учениците да разберат основните функции. По този начин обединяваме 2 основни стъпки от разработката на системата, като спестяваме време и получаваме нови сценарии за работа с приложенията, които служат за тестови ситуации.

6. Въведение в експлоатация

След края на всички тестове системата се въвежда за експлоатация, като първите 3 месеца се използва и системата - метода, който е използван до момента. Това е с цел да се избегнат грешки при отчитането на работниците в ДСП. Ако след 3-те месеца няма разликите в отчетите направени на ръка и тези, които са генерирани от системата, то тогава ще остане само системата.

4.3 Ниво на сложност на проекта – основни проблеми при реализацията на поставените цели

При разработката на приложението първите моменти, при които срещнахме затруднения бяха при избора на база данни, която да е онлайн и да е безплатна или поненискобюджетна. И изучаването на новата база, която не се учи задълбочено в часовете ни в училище, а само разглеждахме различните NoSQL база данни.

В последствие друга трудност, която срещнахме беше при изработването на отчетите. Поради различната дата за край на работния месец, всеки месец при ДСП, се наложи да се въведе допълнително получаване на входни данни от администратора на данни.

Друга трудност, която срещнахме беше достъпа до личните данни на

учителите и учениците, с които трябваше да напълним нашата база данни. За да получим имената и имейлите на учениците и учителите, трябваше да попълним декларация, в която да посочим точно за какво ще ги ползваме.

4.4 Логическо и функционално описание

Като основен език при разработката на системата е използван езика JavaScript и Node JS(Runtime Environment).

За Front-end Framework на мобилното приложение е използван **React Native** (създаден от Meta/Facebook и използван от Facebook, Instagram, Uber и други) и **Expo** – допълнителен пакет за изработването на cross-platform приложения. За конвертирането на React Native синтаксис (ECMAScript 6), в съвместима на различните машини версия на JavaScript, е използван транскомпилятора **Babel**.

За направата на десктоп приложението са използвани **React JS** и **Electron JS** като Front- end Frameworks. React JS е JavaScript Framework използван за изработването на интерактивен потребителски интерфейс и уеб приложения със значително по-малко код. React JS също използва Babel. Electron JS е един от най-добрите избори за преминаването от уеб в десктоп апликация. Използван от големи компании като WhatsApp, Spotify, Discord и Twitch и с помощта на Chromium (проект с отворен код създаден от Google, чийто код служи като основа на Microsoft Edge, Opera и други) този Framework се е доказал през годините.

Пакетът **Dotenv** позволява създаването и използването на .env файл, който съхранява и защитава пароли, API ключове и друга важна информация под формата на key-value pair(KVP).

За база данни е избрана **Firebase** (NoSQL). Тъй като мащабът на системата не е голям и нужната информация е в малки количества, сме заложили на бързината, която NoSQL базата данни предоставя. Използва се **Firebase SDK** (Software Development Kit), който улеснява използването на **API** (Application Programming Interface) като комуникация между двете приложения и базата данни. Firebase има много други преимущества освен бързина и лесна връзка. **Firebase Authentication** предоставя методи за създаване и управление на акаунти чрез имейл и парола. **Firebase Cloud Functions** е serverless framework, който позволява автоматичното изпълнение на предварително зададени функции, съхранени в Google Cloud. **Cloud Firestore Security Rules** са правила, съхранени в Google Cloud, които контролират достъпа до информация в базата данни.

4.5 Реализация

При реализацията на проекта работата е разделена главно на 2 части:

- Front-end - UI/UX (User Interface/User Experience)
- Back-end - Database, Database connection, APIs

Първата стъпка е да се създадат двете приложения, които са празни, без никакво съдържание и готови за разработка.

В основата на мобилното приложение стои React Native и Expo. Чрез Expo създаваме ново приложение. За визуализацията на приложението са използвани софтуерите Android Studio (официалната интегрирана среда за разработка на операционната система Android на Google) и Xcode (интегрирана среда за разработка на Apple за iOS, macOS и други). Те предоставят емулятори за Android и iOS, чрез които мобилното приложение бива визуализирано.

При десктоп приложението се използват React JS и Electron JS. Чрез React JS се изготвя локално хостнат уебсайт. След това чрез Electron JS създаваме приложение и задаваме URL линка към локално хостнатия сайт като „entry point“ (местоположение в програмата, от която започва изпълнението на кода). Стартираме Electron JS, десктоп приложението се отваря и в него се зарежда уебсайта. След приключването на тази стъпка приложенията са готови за разработка.

Стъпка номер две е изработката на интерфейс и свързването му с базата данни. Като за начало Front-end разработчикът изготви бърз и лесен интерфейс, без стилистична обработка (User experience design). Това са различните прозорци и навигацията между тях, бутони, полета за въвеждане на текст, компоненти съдържащи информация от базата данни. През това време Back-end разработчик създаде база данни съдържаща колекции и документи, както и няколко акаунта с различни нива на достъп.

След приключване на тази стъпка, двамата разработчици започнаха следващите си задачи. Front-end разработчикът премина към изготвянето на дизайн, като избра цветовете, темите, сенките анимациите, подредбата и големината на елементите, фон, изображения, шрифт и други. Back-end разработчикът създаде връзката между интерфейса и базата данни. Създаде и функциите за извличане, актуализиране, създаване и изтриване на информация от базата данни, нужни на интерфейса за неговата функционалност.

След като дизайнът бе избран, а функционалността готова, преминахме напред и Front-end разработчик започна сглобяването на приложенията, а Back-end разработчикът създаде Firebase Cloud Functions, чието предназначение е автоматичното изпълнение напредварително зададени функции, като всекидневно актуализиране на датата, всекидневно запазване и нулиране на дадени полета и други.

Също така състави правила в Cloud Firestore Security Rules, които предоставят сигурност на информацията в базата данни.

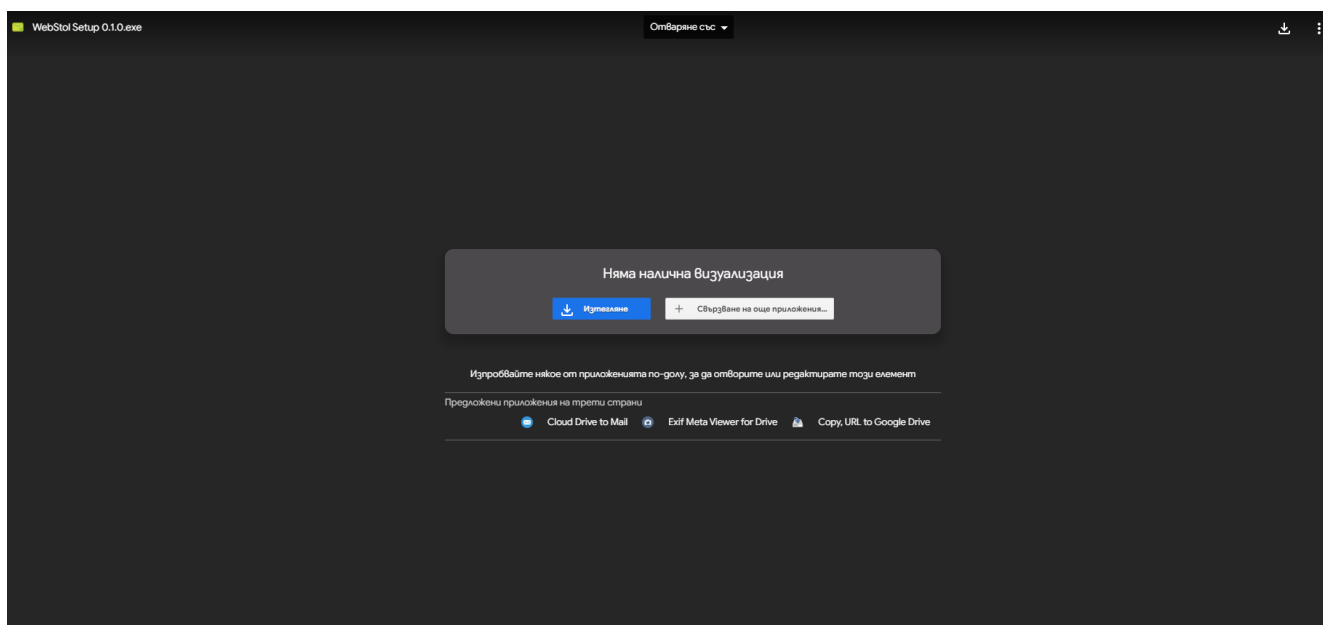
След приключване на тези стъпки приложенията бяха готови за конфигуриране и създаването на инсталационни файлове. След като се инсталират за проверка на инсталационните файлове, може да преминем към последни щрихи. Успоредно с тях приложенията се тестват с цел откриване на бъгове, грешки в кода и несъответствие на информация, като проблемите незабавно се отстраняват.

4.6.Описание на приложението

Инсталиране на настолното приложение

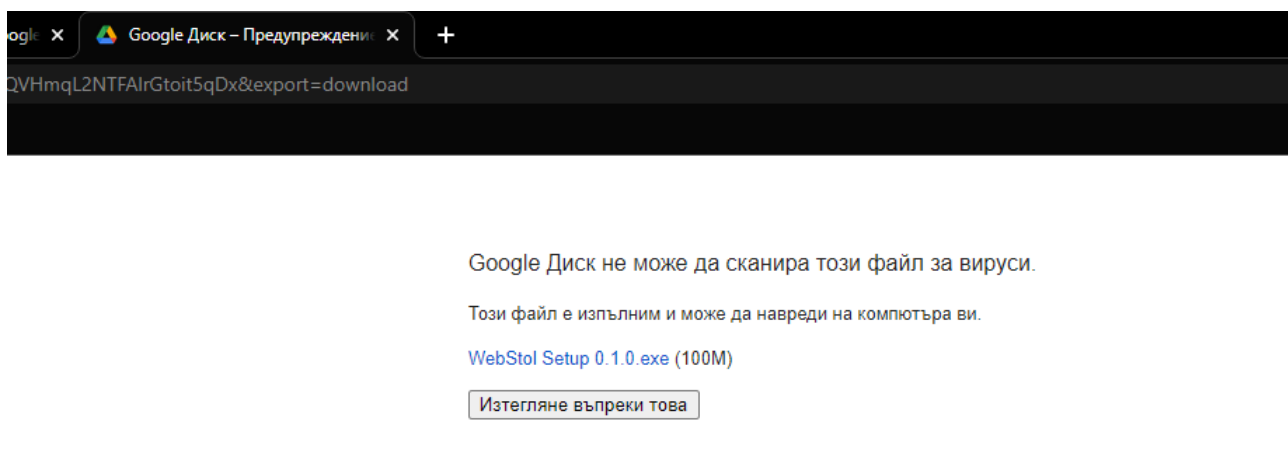
Изпраща се линк за сваляне от Гугъл Диск, който ви праща към фиг. 4.6.1.

Фиг. 4.6.1

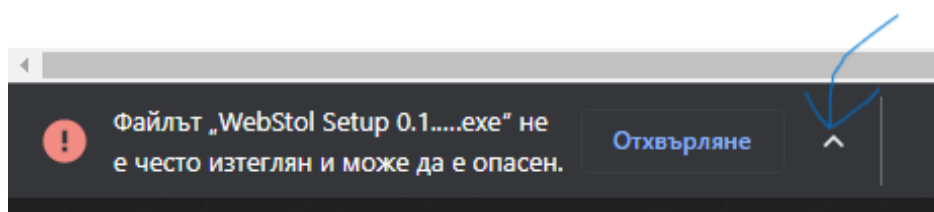


След като кликнем върху “Изтегляне” ще бъдем пренасочени към фиг. 4.6.2.

Фиг. 4.6.2.

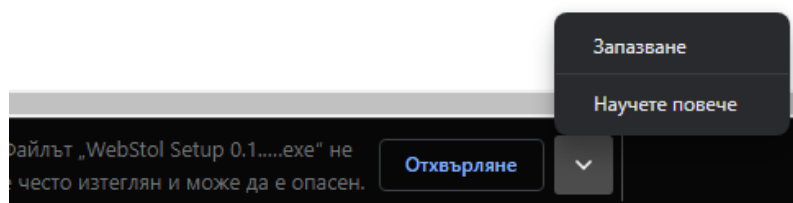


След като изтеглянето завърши, вероятно е да видите следното фиг. 4.6.3.
Фиг. 4.6.3.



В такъв случай кликваме на мястото, посочено от стрелката и избираме “Запазване”
- фиг. 4.6.4

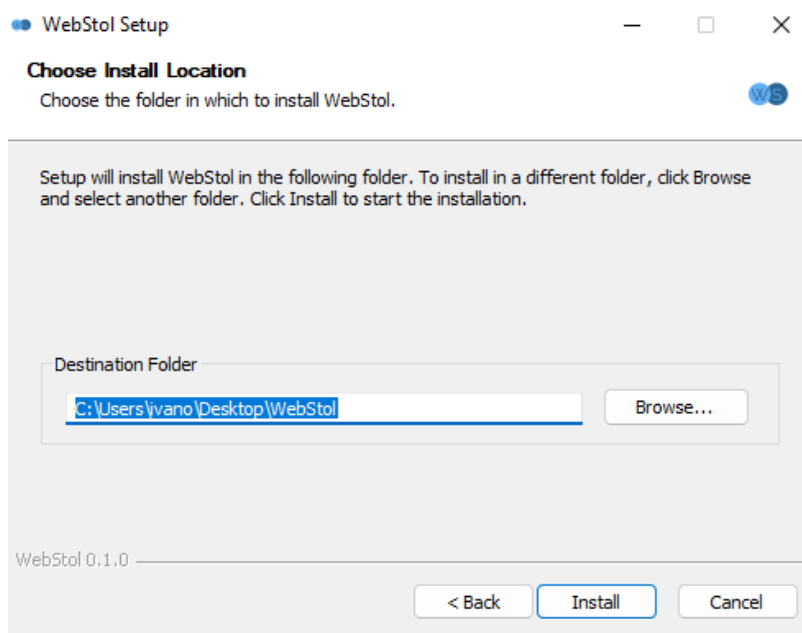
Фиг.4.6.4



Следва да го отворим, за да започнем процеса на изтегляне.

Избираме къде да бъде запазено приложението и го инсталираме фиг. 4.6.5

Фиг. 4.6.5.



Използване на настолното приложение

След инсталирането на .Ехе файла е необходимо да се стартира от иконата за старт. Фиг.4.6.6.

Фиг.4.6.6.



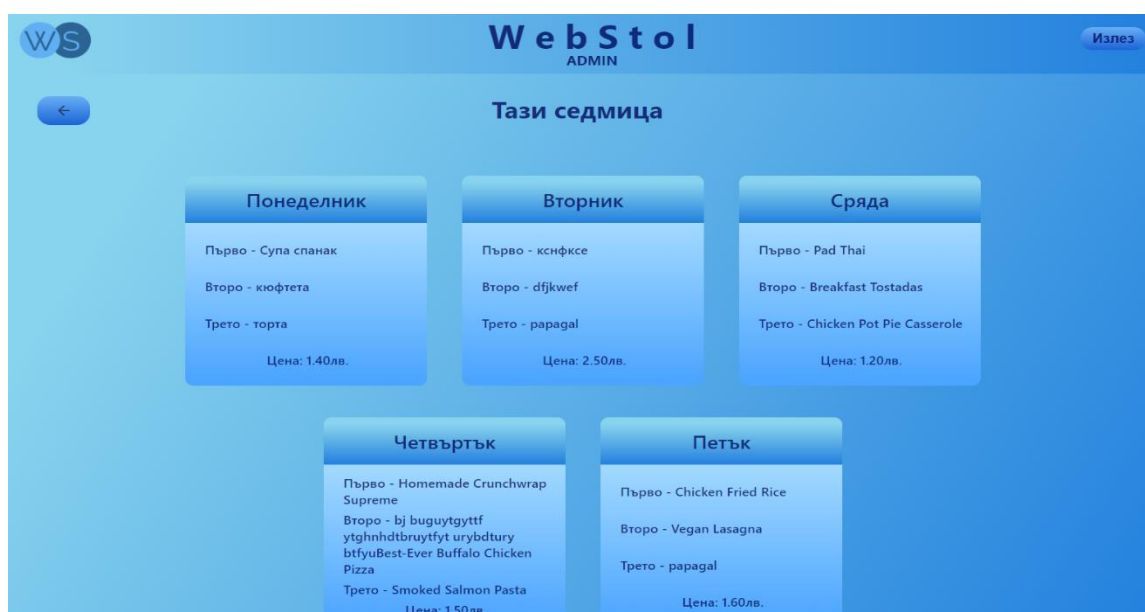
След стартирането се появява диалогов прозорец за вход в системата фиг. 4.6.7.

Фиг. 4.6.7.



Входният прозорец е еднакъв за всички потребители, като в зависимост от нивото на достъп има различни права за работа в приложението. Хоум прозорецът за администратора фиг.4.6.8. се състои от 3 подпрозореца, които са съответно и връзки към тях, при поставяне на курсора върхи всеки един от тях фонът мусе замъглява и излиза съответно кратка информация, затова което ще видим в диалоговия прозорец.

Фиг.4.6.8.



Диалоговите прозорци за другите нива на достъп са идентични в зависимост от правата им.

Меню прозореца.

-При кликване върху големия прозорец ще бъдем препратени към менюто от фиг. 4.6.8. Тук можем да разберем какво е менюто за тази седмица и каква е цената за съответния ден. Стрелката в горен ляв ъгъл ще ни върне към Хоум страницата при натискане.

След едно скролване стигаме до частта от страницата в която можем да въведем менюто за всеки отделен ден през следващата седмица фиг. 4.6.9. При натискане на бутона “Запази меню” всяка промяна ще бъде запазена. Освен въвеждане и промяна на цялото меню, ако сме въвели едно ново хапване или цена и всички други полета са празни, при запазване ще бъде променена стойността само избраното от нас поле (пример: Понеделник-Второ) без да засяга останалата част от менюто.

Фиг. 4.6.9.

The screenshot displays a menu selection interface with five panels for the days of the week: Понеделник (Monday), Вторник (Tuesday), Сряда (Wednesday), Четвъртък (Thursday), and Петък (Friday). Each panel contains four input fields labeled Първо (First), Второ (Second), Трето (Third), and Цена (Price). Below the panels is a button labeled Запази меню (Save menu).

Под бутона “Запази меню” се намират два отделни прозореца за промяна на наличният хляб и на текущата му цена фиг. 4.6.10.

Фиг. 4.6.10

The screenshot shows a confirmation window with a button labeled Запази меню (Save menu) at the top. Below it are two input fields: Цена на хляба - 2.50 (Bread price - 2.50) and Брой - 4.00 (Quantity - 4.00). Each field has a Запази (Save) button below it. At the bottom, there is a logo with the letters 'WS' and the text Copyright © 2023 KPGI.

Списъкът с ученици:

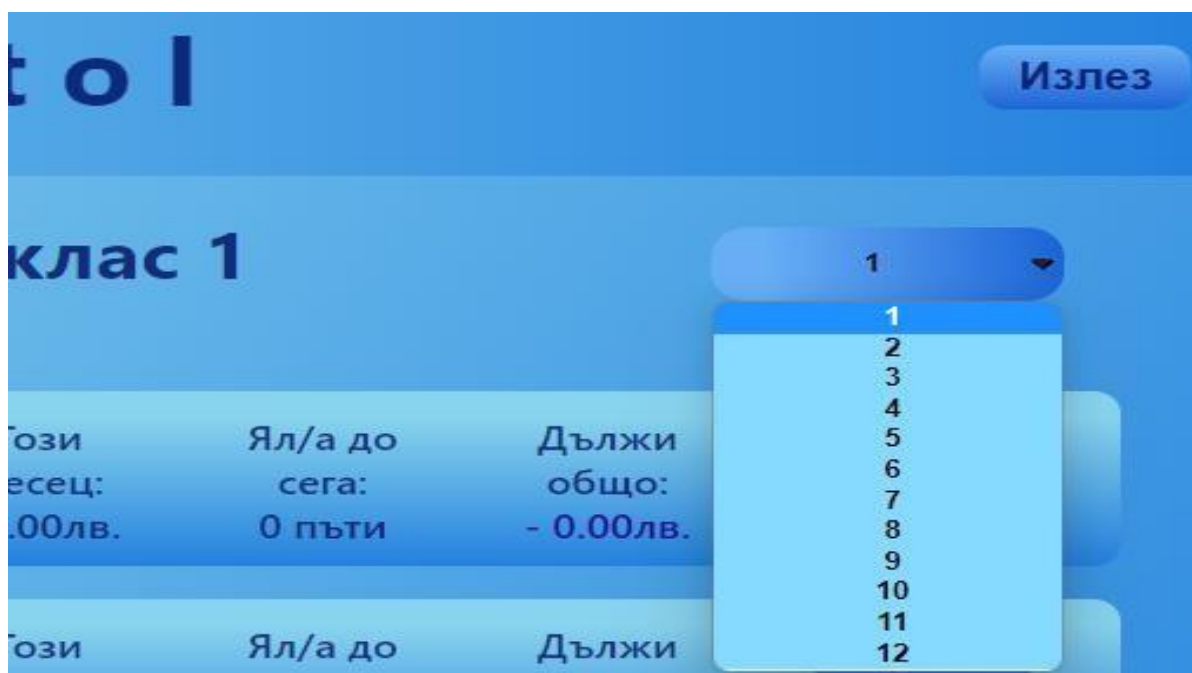
- При кликане върху прозореца в долният десен ъгъл ще бъдем отведени към списъка с ученици фиг. 4.6.11.

Фиг. 4.6.11.

№	Име	Няма да яде!	Да	Не	+	-	Този месец:	Ял/а до сега:	Дължи общо:	Платено	Неплатено
№ 1	Илианка Стоянова Михайлова	Няма да яде!	Да	Не	+	-	- 0.00лв.	0 пъти	- 0.00лв.	Платено	Неплатено
№ 1	Александър Насков Меслеков	Няма да яде!	Да	Не	+	-	- 0.00лв.	0 пъти	- 0.00лв.	Платено	Неплатено
№ 2	Виолета Николаева Йончева	Няма да яде!	Да	Не	+	-	- 0.00лв.	0 пъти	- 0.00лв.	Платено	Неплатено
№ 2	Лиляна Нешова Бучкова	Няма да яде!	Да	Не	+	-	- 0.00лв.	0 пъти	- 0.00лв.	Платено	Неплатено
№ 3	Георги Валентинов Желязков	Няма да яде!	Да	Не	+	-	- 0.00лв.	0 пъти	- 0.00лв.	Платено	Неплатено
№ 3	Михаел Толев Тороманов	Няма да яде!	Да	Не	+	-	- 0.00лв.	0 пъти	- 0.00лв.	Платено	Неплатено
№ 3	Десьо	Няма да яде!	Да	Не	+	-	- 0.00лв.	0 пъти	- 0.00лв.	Платено	Неплатено

Можем да изберем от кой клас да бъдат показани ученици фиг. 4.6.12.

Фиг. 4.6.12.



От бутоните “Да” и “Не” фиг. 4.6.12 можем да запишем или отпишем съответния ученик Бутоните “+” “-”

Фиг.4.6.13.

Ученици от клас 1					
№ 1	Божидара Христова Лукова	Не е записан/а	Да Не	+ -	Този месец: 0.00лв.
№ 2	Бориана Динчова Азманова	Не е записан/а	Да Не	+ -	Този месец: 0.00лв.
№ 3	Велимир Цветомиров Васев	Не е записан/а	Да Не	+ -	Този месец: 0.00лв.

Бутоните “Платено” и “Неплатено” фиг. 4.6.14. ни позволяват да следим дали ученикът е заплатил дължимото. Ако не е, той не може да се записва да яде от ученическия интерфейс, нито ние можем да го запишем от учителския или тук докато не си заплати дължимото.

Фиг. 4.6.14

Клас 1			
Ял/а до сега:	Дължи общо:	Платено	
0 пъти	0.00лв.	Неплатено	
Ял/а до сега:	Дължи общо:	Платено	
0 пъти	0.00лв.	Неплатено	

Месечен отчет:

- При кликане върху прозореца в долен ляв ъгъл ще се озовем в месечния отчет фиг.4.6.15. Тук можем да зададем нова дата за месечен отчет и начална/крайна дата за пауза на системата. Целта на паузата е децата да не могат да се записват през ваканции, извънредни ситуации и тн. Ако зададем нова начална дата, достъпът се спира веднага щом

дойде тази дата и се възобновява щом текущата дата съвпада със зададената от наскрайна дата на паузата.

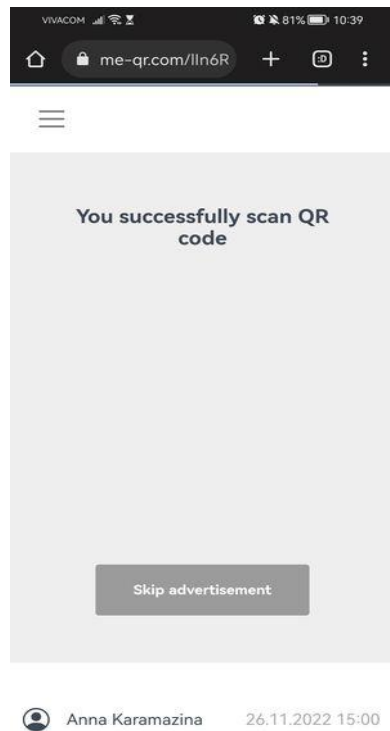
Фиг. 4.6.15.

The screenshot displays the 'WebStol ADMIN' web application interface. At the top, there is a header bar with the 'WS' logo on the left, the text 'WebStol ADMIN' in the center, and an 'Излез' (Logout) button on the right. Below the header, a blue sidebar contains a back arrow button. The main content area features three distinct form panels. The top-left panel, titled 'Пауза-начална дата' (Pause start date), shows 'Текуща дата - 100' (Current date - 100) and a 'Дата:' input field with a 'Промени дата' (Change date) button below it. The top-right panel, titled 'Пауза-крайна дата' (Pause end date), also shows 'Текуща дата - 100' and a 'Дата:' input field with a 'Промени дата' button. The bottom-center panel, titled 'Дата за месечен отчет' (Date for monthly report), shows 'Текуща дата - 100' and a 'Дата:' input field with a 'Промени дата' button. The footer of the page includes the 'WS' logo and the text 'Copyright © 2023 KPGI'.

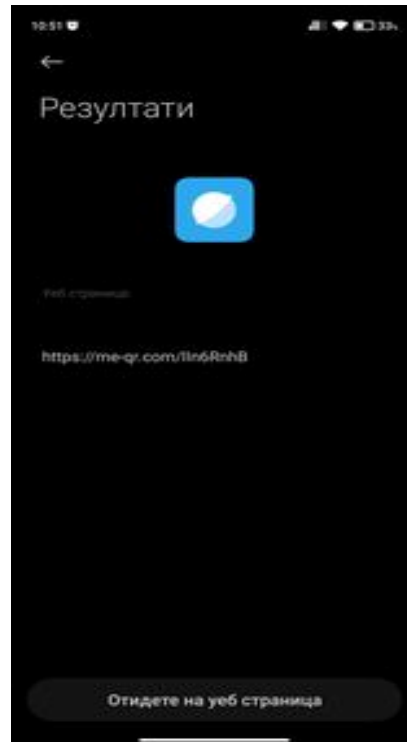
Инсталиране на мобилното приложение за Андроид

След като сканираме QR-кода фиг.4.6.16 и кликнем “Skip advertisement” ще влезем в сайта да приключим изтеглянето фиг.4.6.17

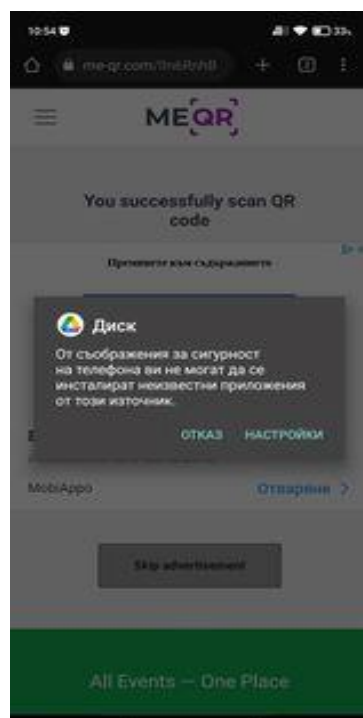
Фиг. 4.6.16



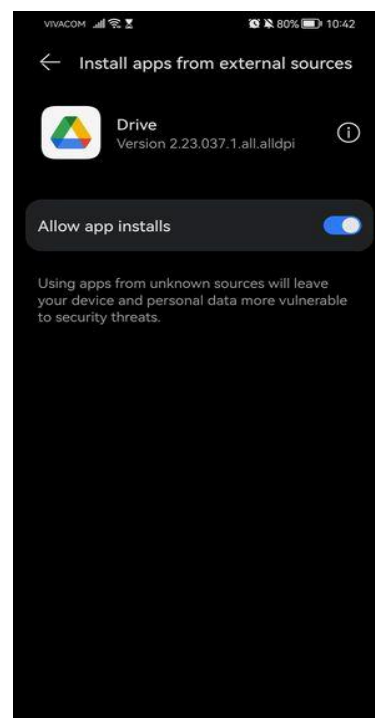
Фиг. 4.6.17



Фиг. 4.6.18



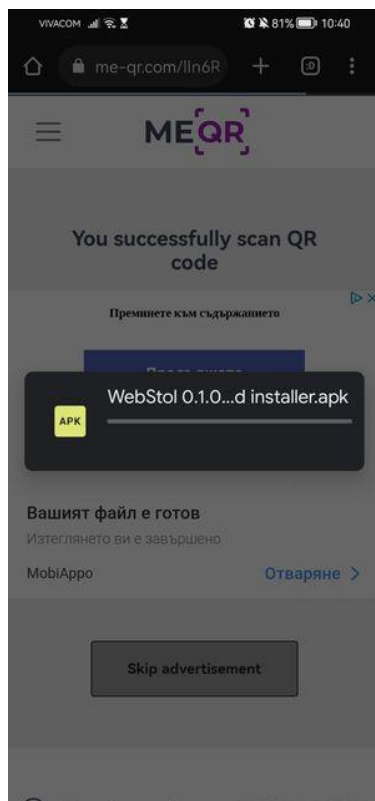
Фиг. 4.6.19



Има вероятност Гугъл диск да поиска разрешение за инсталиране на неизвестни

приложения, това се вижда на фиг. Фиг. 4.6.18 и Фиг. 4.6.19. След което трябва да започне инсталирането - Фиг. 4.6.20

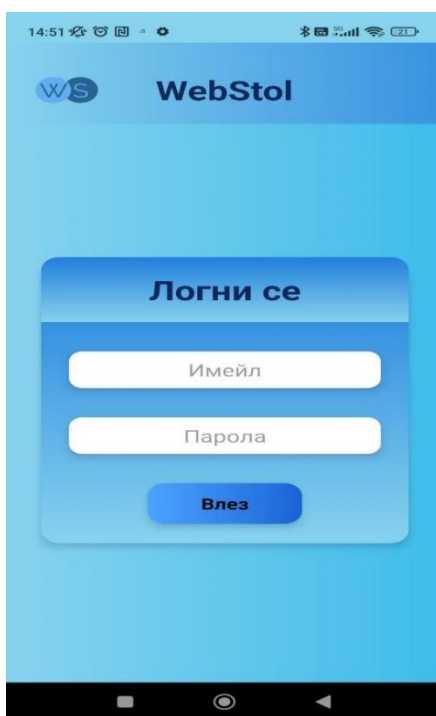
Фиг. 4.6.20



Работа с мобилното приложение

След стартирането ще бъдем отведени към прозореца за вход в системата Фиг. 4.6.21.

Фиг. 4.6.21



В зависимост дали имейла, с който сме влезли има зададена роля на ученик или на

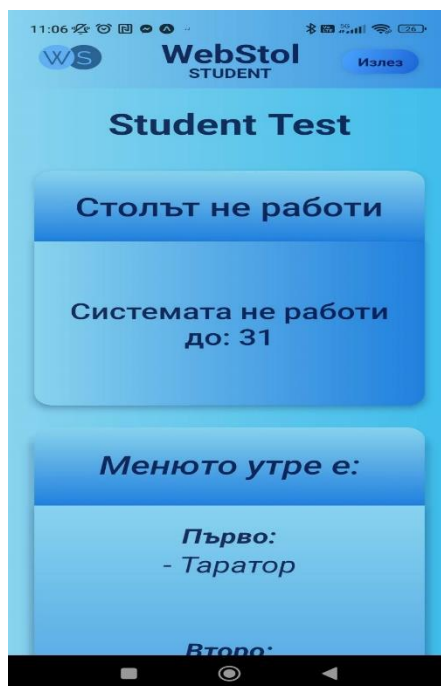
учител ще бъдем отведени към различен интерфейс:

Ученически интерфейс

Фиг.4.6.22.



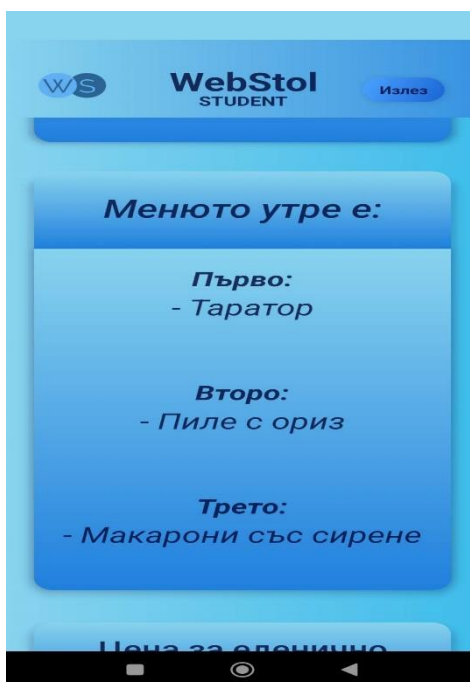
Фиг.4.6.23



Тук ученикът ще види своето име и ще може да се запише за следващия ден, като състоянието на “Не е записан/а” “Записан/а” се променя почти мигновено (Фиг.4.6.22). Ако системата е поставена на пауза, прозорецът ще бъде заменен (Фиг.4.6.23).

На Фиг. 4.6.24 можем да видим менюто за утре.

Фиг. 4.6.24.



Цената на менюто за дадения ден се вижда на Фиг.4.6.25.

Фиг.4.6.25.



Когато датата за плащане е определена от стола и е въведена от администратор ще се изпише в прозорчето за цената - фиг. 4.6.25.

Учителски интерфейс

Тук включваме всичко от ученическия интерфейс плюс списък с учениците и долен таб бар за навигация между списъка и началната страница фиг. 4.6.26.

Фиг. 4.6.26.



В списъка виждаме учениците от посочения клас, сортирани по номер в класа. Учителят може да вижда учениците само от неговия клас фиг. 4.6.27.

Фиг. 4.6.27

The screenshot displays the 'WebStol TEACHER' app interface. At the top, there is a header with a 'WS' logo, the text 'WebStol TEACHER', and an 'Излез' (Logout) button. Below the header, the title 'Ученици от клас 1' (Students from Class 1) is prominently displayed. The main content area lists three students, each with a unique identifier and name: №1: Божидара Христова Лукова, №2: Бориана Динчова Азманова, and №3: Велимир Цветомиров Васев. For each student, there is a section titled 'Ще яде ли?' (Will they eat?) with two buttons: 'Да' (Yes) and 'Не' (No). To the right of these buttons, the 'Статус:' (Status) is shown as 'Не е записан/а' (Not registered) and the 'Дължи: 0.00лв.' (Owed: 0.00 BGN) is displayed. At the bottom of the screen, there is a navigation bar with two icons: a house icon labeled 'Начало' (Home) and a document icon labeled 'Лист' (List). The entire interface is set against a light blue background with rounded rectangular elements.

№	Име	Ще яде ли?	Статус	Дължи
№1:	Божидара Христова Лукова	<input type="button" value="Да"/> <input type="button" value="Не"/>	Не е записан/а	0.00лв.
№2:	Бориана Динчова Азманова	<input type="button" value="Да"/> <input type="button" value="Не"/>	Не е записан/а	0.00лв.
№3:	Велимир Цветомиров Васев	<input type="button" value="Да"/> <input type="button" value="Не"/>	Не е записан/а	0.00лв.

4.7.Заклучение

В заключение имаме една работеща система, която е свързана с база данни и се ползва от над 200 потребители. Всеки може да дойде да я види и да я използва. Според нас крайната цел на диагностицирания проблем е достигната, а за бъдещо развитие това, което ще направим е:

- Приложенията за телефони да бъдат качени в официалните магазини за приложения съответно за Android и за IOS;
- При развитие на дейността на ДСП Копривщица ще можем да добавим още едно ниво за потребителски достъп CLIENT, което да е за достъп до всички граждани и гости на града;
- Да направим опция за чат и размяна на съобщения, между отделните класове или групи ученици, както и директно с администратора към стола;

Системата решава първоначалния открит от нас проблем.