

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**  
( Н И У « Б е л Г У » )

ИНСТИТУТ ИНЖЕНЕРНЫХ И ЦИФРОВЫХ ТЕХНОЛОГИЙ  
КАФЕДРА ИНФОРМАЦИОННЫХ И РОБОТОТЕХНИЧЕСКИХ СИСТЕМ

**Автоматизированная система контроля и планирования рациона  
питания**

Выпускная квалификационная работа  
обучающегося по направлению подготовки 09.03.02 Информационные  
системы и технологии  
очной формы обучения, группы 12001508  
Бурлакова Николая Вячеславовича

Научный руководитель  
Ст. преподаватель  
Федоров Вячеслав Игоревич

БЕЛГОРОД 2019

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1 Аналитическая часть.....	6
1.1 Обзор принципов и методов, используемых в диетологии.....	6
1.2 Анализ существующих аналогов системы.....	13
1.1.1 Веб-сайт FatSecret.....	13
1.1.2 Веб-сайт MyfitnessPal.....	14
1.1.3 Сравнительный анализ прочих систем.....	16
1.3 Определение функций системы.....	17
2 Моделирование системы.....	18
2.1 Обоснование выбора базы данных продуктов питания.....	18
2.2 Обоснование выбора программных средств.....	19
2.3 Проектирование базы данных.....	22
3 Реализация системы.....	28
3.1 Разработка базы данных.....	28
3.2 Разработка интерфейса системы.....	33
3.3 Программная реализация системы.....	37
3.4 Тестирование и отладка системы.....	43
3.5 Метод SWOT-анализа.....	50
ЗАКЛЮЧЕНИЕ.....	53
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	55
ПРИЛОЖЕНИЕ А.....	58
ПРИЛОЖЕНИЕ В.....	59

## ВВЕДЕНИЕ

Цифровые технологии в настоящее время занимают особое место в жизни человека, облегчая всевозможные действия в процессе его жизнедеятельности. Поддержание здорового образа жизни не является исключением.

За последние 12 лет число россиян, периодически занимающихся спортом, выросло в полтора раза. По подсчетам всероссийского центра изучения общественного мнения (ВЦИОМ) в 2006 году таких людей насчитывалось около 38 процентов, а к 2018 году 60 процентов [30]. Как в профессиональном спорте, так и в поддержании здорового образа жизни важно придерживаться правильного питания. Правильное питание включает в себя грамотное планирование и слежение за рационом питания, знание основных принципов диетологии, контроль над потреблением пищи и воды, потребление определенных продуктов, содержащих жизненно необходимые микро-, макроэлементы, витамины и минералы и т.д.

Отказ от ведения правильного питания, а также неграмотное составление рациона и следование неправильным диетам приводит к многочисленным заболеваниям (состояниям), таким как заболевания желудочно-кишечного тракта, аллергия на пищевые продукты, ожирение, сердечно-сосудистые заболевания, сахарный диабет, повышенное артериальное давление, депрессия и прочие. По данным сборника «Рацион питания населения. 2013» 83% россиян согласны с утверждением о том, что пища сильно влияет на вероятность возникновения заболеваний, например, сердечно-сосудистых и 44,9% имеют одно или несколько заболеваний (состояний), связанных с питанием.

В условиях дефицита времени не каждый человек имеет возможность вникать в принципы диетологии и самостоятельно выполнять остальные рекомендации по поддержанию правильного питания. К тому же, в

различных информационных ресурсах присутствует множество недостоверной или неподтвержденной информации, что может повлечь за собой негативные последствия для здоровья человека. По мнению 80% мужчин и женщин различных возрастных категорий, существует настолько много рекомендаций относительно здорового питания, что трудно понять, что из этого является достоверной информацией [30].

Для решения этой проблемы имеется множество приложений для мобильных устройств, планшетных и настольных компьютеров, а так же веб-сайтов. Данные приложения направлены на то, чтобы упростить пользователю процесс поддержания правильного питания и в каждом из них поддерживаются разные функции. Среди них можно выделить планировщики рациона, информационные справочники о продуктах, приложения контроля питания, контроля потребления воды и другие.

Несмотря на большое количество приложений в данной отрасли, они во многом оказываются неудобны из-за сжатого набора функций, а также редко предоставляют возможность полного, комплексного контроля рациона питания. Большинство подобных приложений является инструментом для отслеживания калорийности съеденной пищи, а также базовых биологически значимых элементов, не предоставляя полной информации о содержании в пище всех макро-, микроэлементов и витаминов.

Несмотря на то, что основная масса приложений, облегчающих процесс поддержания правильного питания, рассчитаны на мобильные устройства, было принято решение о создании веб-сайта, так как сайт способен вместить необходимое количество функций и при этом иметь интуитивно понятный интерфейс. Адаптивный дизайн позволяет одинаково удобно пользоваться сайтом на устройствах с дисплеями любого размера.

Все вышеперечисленные факторы определяют актуальность данной темы. Для решения данной проблемы были поставлены цели и задачи выпускной квалификационной работы.

Целью работы является совершенствование процесса планирования и контроля питания человека посредством разработки системы.

Достижение указанной цели возможно посредством решения следующих основных задач:

- изучение и выбор методов диетологии, используемых для расчета рекомендуемой суточной нормы потребления;
- выбор базы данных продуктов питания;
- анализ существующих аналогов;
- определение функций системы контроля и планирования питания;
- изучение и выбор средств для разработки;
- проектирование системы контроля и планирования питания;
- разработка системы контроля и планирования питания;
- тестирование системы контроля и планирования питания.

Выпускная квалификационная работа выполнена на 56 страницах без приложения, и 70 страницах с приложениями, содержит 33 рисунка, 12 таблиц и 2 приложения.

## 1 Аналитическая часть

### 1.1 Обзор принципов и методов, используемых в диетологии

Проблема здорового питания – сложная и комплексная проблема, требующая обширных знаний и навыков в различных областях науки и практики. Полноценность питания в значительной мере определяет здоровье и благоприятное самочувствие, как и лечение уже появившихся болезней невозможно без продуктов, полезных для организма.

Для разработки системы контроля и планирования рациона питания необходимо ознакомиться с основными принципами диетологии, методами, используемыми для расчета различных индивидуальных данных и т.д.

В XVIII-XIX веках успехи химии и смежных наук позволили разработать представления о питательных веществах, содержащихся в пищевых продуктах, об энергетической ценности пищи, о физиологии пищеварения. Накапливались знания об основных компонентах пищи – белках, жирах, углеводах; эти вещества рассматривались, прежде всего, как источник энергии для организма. В конце XIX и в XX веке становится понятным, что продукты питания должны содержать много других веществ, которые также совершенно необходимы для нормальной жизнедеятельности. Поэтому предметом всестороннего изучения стали аминокислоты и незаменимые жирные кислоты, витамины и витаминоподобные вещества. В последние годы исследователи все больше узнают о важной роли для человеческого организма отдельных химических элементов – биоэлементов [15].

Витамины, биоэлементы и другие биологически активные вещества не представляют для организма энергетической ценности, поскольку не являются, подобно жирам или углеводам, источником калорий. Но эти биоактивные вещества, содержащиеся в пище в незначительных количествах,

обеспечивают регуляцию важнейших жизненных функций и нормальное протекание всех жизненных процессов. Это обуславливает важность данных пищевых компонентов для человека [16].

Одна из наиболее распространенных причин нарушений обмена веществ и развития патологий, связанных с данным явлением – неправильное питание. Обычно это стойкие нарушения витаминного, минерального и других видов обмена, что подчеркивает важность контроля за всеми жизненно важными компонентами продуктов питания. Данные компоненты в полном объеме будут предоставляться пользователям системы, предоставляя возможность отслеживать их содержание в каждодневном рационе.

Возникла необходимость обратиться к нутрициологии для изучения необходимой информации о здоровом питании человека.

Нутрициология – это наука о пище и питании, о продуктах питания, о пищевых веществах и других компонентах, содержащихся в этих продуктах, об их действии и взаимодействии об их потреблении, усвоении, расходовании и выведении из организма, об их роли в поддержании здоровья или возникновении заболеваний [29] .

Все пищевые продукты состоят из необходимых организму питательных веществ. Пищевые вещества, или нутриенты – это органические или неорганические вещества, входящие в состав продуктов. Организм использует пищевые вещества для построения и обновления клеток и тканей, для регуляции биохимических и физиологических функций, а также для получения энергии, необходимой для функционирования различных органов, выполнения физической и умственной работы, поддержания температуры тела [31]. Пищевые вещества разделяют на макро - и микронутриенты.

Макронутриенты – это основные пищевые вещества – белки, жиры, углеводы, которые при окислении дают организму энергию, необходимую для выполнения всех его функций. К основным компонентам питания, необходимым организму в больших количествах (от 1,5л ежедневно), следует относить и питьевую воду. В разрабатываемой системе планируется

использовать информацию о макроэлементах, полученную из базы данных продуктов питания.

Микроэлементы – пищевые вещества, представленные витаминами, биоэлементами, некоторыми минеральными веществами и т.д. [31]. Аналогично макроэлементам информацию о микроэлементах должна быть отражена в базе данных разрабатываемой системы контроля и планирования рациона питания.

Рекомендуемая суточная норма потребления микро- и макроэлементов представлена в таблице 1.

Таблица 1 – Суточная норма потребления микро- и макроэлементов

Название элемента	Рекомендуемая суточная норма
Железо	10 мг. для мужчин и 20 мг. для женщин
Цинк	15 мг.
Медь	1,5 – 3 мг.
Кобальт	0,04-0,07 мг.
Марганец	2 – 5 мг.
Селен	0,04-0,07 мг.
Фтор	0,5 – 0,8 мг.
Йод	0,1 – 0,2 мг.
Кальций	0,8 – 1 мг.
Фосфор	1,6 – 2 мг.
Магний	0,5 – 0,9 мг.
Натрий	5 – 10 мг.
Хлор	4 – 6 мг.
Сера	0,5 – 0,8 мг.



Основной энергетический материал дают организму жиры, белки и углеводы. Считают, что 1г. белков пищи обеспечивают организму 4,1 ккал(17,17 кДж), 1г. жиров – 9,3 ккал (38,96 кДж) и 1г. углеводов – 4,1 ккал. Таким образом, зная химический состав пищи, можно подсчитать калорийность любого продукта или диеты.

В соответствии с современными представлениями суточный рацион здорового человека должен иметь соотношение белков, жиров и углеводов 1:1,2:4. Калорийность пищевого рациона должна соответствовать энергетическим затратам организма. Здоровому человеку от 12 до 17% энергии следует получать за счет белков, от 25 до 35% - за счет жиров и от 50 до 55% - за счет углеводов.

Витамины – незаменимые факторы питания, необходимые для обеспечения всех жизненно важных функций организма. Отслеживание уровня витаминов в потребляемой пище является очень важным аспектом здорового образа жизни. Нехватка витаминов или излишнее их потребление может привести к различным негативным последствиям для организма, таким как гиповитаминоз, авитаминоз, гипервитаминоз.

Рекомендуемая суточная норма потребления витаминов представлена в таблице 2.

Таблица 2 – Суточная норма потребления витаминов

Название витамина	Рекомендуемая суточная норма
А	1 мг.
В1	1 – 2 мг.
В2	1,5 – 2,4 мг.
В6	2 мг.
Д	2,5 мкг.
Е	10 мг.
С	75 – 150 мг.

Потребность человека в отдельных витаминах зависит от возраста, состояния здоровья, характера деятельности, времени года, полноценности питания. Физическое напряжение и интенсивная умственная деятельность сопровождаются повышенным расходом ряда витаминов. Поэтому величина среднесуточной потребности в отдельных витаминах значительно варьирует.

Для косвенной оценки массы человека необходимо знать его индекс массы тела. Индекс массы тела (ИМТ) – величина, позволяющая оценить степень соответствия массы человека и его роста. ИМТ является важным показателем при определении показаний для необходимости лечения.

Индекс массы тела рассчитывается по формуле 1:

$$I = \frac{m}{h^2}, \quad (1)$$

где  $m$  — масса тела в килограммах,  $h$  — рост в метрах, и измеряется в кг/м<sup>2</sup>.

В соответствии с рекомендациями ВОЗ разработана интерпретация показателей ИМТ, представленная в таблице 3.

Таблица 3 – Интерпретация показателей ИМТ

<b>Индекс массы тела</b>	<b>Соответствие между массой человека и его ростом</b>
16 и менее	Выраженный дефицит массы тела
16 – 18,5	Недостаточная (дефицит) масса тела
18,5 – 24,99	Норма
25 – 30	Избыточная масса тела (предожирение)
30 – 35	Ожирение
35 – 40	Ожирение резкое
40 и более	Очень резкое ожирение

Для расчета необходимого суточного потребления калорий были найдены 2 формулы: формула Харриса-Бенедикта, формула Маффина-Джеора.

Формула Харриса-Бенедикта была выведена в 1919 году. В настоящее время считается наиболее точной по отношению к людям, ведущим физически активный образ жизни. Формула состоит из двух основных показателей: базальный метаболизм и активный метаболизм. Базальный метаболизм для мужчин представлен на формуле 2, а для женщин на формуле 3.

$$BMR = 447,593 + (9,247 * m) + (3,098 * h) - (4,330 * a), \quad (2)$$

$$BMR = 88,362 + (13,397 * m) + (4,799 * h) - (5,677 * a), \quad (3)$$

где  $m$  — масса тела в килограммах,  $h$  — рост в сантиметрах,  $a$  — возраст в годах.

Активный метаболизм по формуле Харриса-Бенедикта определяется в зависимости от того, какой образ жизни ведет человек:

- сидячий образ жизни – 1,2;
- умеренная активность (легкие физические нагрузки либо занятия 1-3 раза в неделю) – 1,375;
- средняя активность (занятия 3-5 раз в неделю) – 1,55;
- высокая активность (интенсивные нагрузки, занятия 6-7 раз в неделю) – 1,725;
- экстравысокая активность – 1,9.

Чтобы посчитать суточную норму калорий по формуле Харриса-Бенедикта необходимо умножить 2 величины (базальный метаболизм и активный метаболизм).

Формула основного обмена Маффина-Джеора (или Миффлина-Сан Жеора) – это современный метод расчёта. Формула выведена в 2005 году и по утверждению Американской Диетической Ассоциации (АДА) на сегодняшний день позволяет наиболее точно рассчитать, сколько калорий тратит организм здорового взрослого человека в состоянии покоя.

Расчет базового обмена веществ для мужчин представлен на формуле 4, а для женщин на формуле 5.

$$B00 = 9,99 * m + 6,25 * h - 4,92 * a + 5, \quad (4)$$

$$B00 = 9,99 * m + 6,25 * h - 4,92 * a - 161, \quad (5)$$

где  $m$  — масса тела в килограммах,  $h$  — рост в сантиметрах,  $a$  — возраст в годах.

Для расчета рекомендуемой суточной нормы калорий следует умножить величину базового обмена веществ на значение, соответствующее количеству физической нагрузки, представленному в таблице 4.

Таблица 4 – Активный метаболизм

Количество физической нагрузки	Суточный расход энергии
Минимальные нагрузки (сидячая работа)	$B00 * 1,2$
Небольшая дневная активность и легкие упражнения 1-3 раза в неделю	$B00 * 1,375$
Тренировки 4-5 раз в неделю (или работа средней тяжести)	$B00 * 1,4625$
Интенсивные тренировки 4-5 раз в неделю	$B00 * 1,550$
Ежедневные тренировки	$B00 * 1,6375$
Ежедневные интенсивные тренировки или тренировки 2 раза в день	$B00 * 1,725$
Тяжелая физическая работа или интенсивные тренировки 2-3 раза в день	$B00 * 1,9$

Формула позволяет оценить рекомендуемую суточную норму потребления с учетом уровня активности человека и суточного расхода энергии соответственно, исходя из таблицы 4.

## 1.2 Анализ существующих аналогов системы

В процессе формирования требований к разрабатываемой системе были изучены наиболее популярные программные продукты, оказывающие помощь человеку при поддержании здорового питания.

Выявление недостатков и преимуществ существующих аналогов программных средств позволяет определиться с функциями, которые необходимо реализовать в системе контроля и планирования рациона питания.

### 1.1.1 Веб-сайт FatSecret

Fatsecret – одно из наиболее популярных приложений для планирования и контроля питания. К преимуществам fatsecret можно отнести то, что у данного программного продукта имеется как функциональный сайт, так и мобильное приложение. В данной работе были рассмотрены преимущества и недостатки fatsecret как сайта.

Незарегистрированные пользователи имеют возможность просмотреть пищевую ценность различных продуктов из 18 категорий. В пищевую ценность входят калории, белки, углеводы, жиры, холестерин, клетчатка, натрий и калий. Явным недостатком является отсутствие данных о других нутриентах, витаминах. Также пользователи сайта, не осуществившие вход в систему, могут просмотреть рецепты, публикуемые другими пользователями, а также расход энергии для различных видов деятельности и упражнений.

Для зарегистрированных пользователей fatsecret предлагает следующий набор функций:

- отслеживание изменений собственного веса в виде графика;
- ведение блога;
- наличие объемной базы данных продуктов, включая продукты некоторых крупных производителей;

- создание своей кулинарной книги;
- календарь диеты, позволяющий проследить потребление и сжигание калорий;
- дневник упражнений;
- отслеживание калорий, затраченных во время занятий спортом;
- расчет рекомендуемого суточного количества калорий.

Несмотря на обширный набор функций, fatsecret имеет один явный недостаток, выявленный в ходе использования данного сайта, – отсутствие данных по многочисленным нутриентам, содержащимся в пищевых продуктах. А также рекомендуемая суточная норма потребления рассчитывается только для калорий. Также следует отметить, что при использовании данного сайта могут возникнуть некоторые сложности, интерфейс разрабатываемой системы должен быть интуитивно понятен, чтобы избежать подобных недостатков.

### 1.1.2 Веб-сайт MyfitnessPal

Аналогично предыдущему рассмотренному приложению, MyFitnessPal является одним из наиболее популярных приложений в сфере питания и имеет мобильное приложение и функциональный веб-сайт.

Во многом MyFitnessPal предоставляет схожие функции с предыдущим описанным приложением. На данном сайте предусмотрена возможность просмотра отчетов в виде графиков не только по изменениям веса, но и замерам тела, потреблению калорий, белков, жиров, углеводов и некоторым нутриентам. Также возможен экспорт данных отчетов за указанный отчетный период (до года). MyFitnessPal предоставляет возможность отдельного отслеживания употребления питьевой воды, а также к пищевым компонентам, доступным в предыдущем аналоге, добавлены витамины А и С и содержание сахара. В отличие от fatsecret на данном сайте отсутствует база

данных рецептов других пользователей, но имеется возможность создания собственной кулинарной книги.

Значительным отличием от fatsecret является тот факт, что база данных продуктов питания предусматривает добавление своих продуктов пользователям сайта. Это является одновременно и преимуществом и недостатком myfitnesspal. Добавление продуктов питания пользователями значительно увеличивает объем доступной информации, но, следует учитывать, что пользователь может ввести некорректную (неподтвержденную) информацию, которую в дальнейшем сможет использовать другой человек. Полная информация о составе продуктов питания, подтвержденная разработчиками, отмечена особым образом, но было выявлено, что данный продукт может находиться значительно ниже в списке, нежели пользовательские продукты. Информация о содержании нутриентов в продуктах питания отсутствует в большинстве пользовательских продуктах, а также может не соответствовать действительности, что является явным недостатком предоставления неподтвержденной информации пользователей в публичный доступ.

Как на сайте, так и в приложении MyFitnessPal имеется обычная и премиум (платная) версия. После покупки премиум-версии предлагаются следующие дополнительные функции:

- сводка пищевой ценности;
- анализ блюд;
- экспорт файла;
- цели по калориям для разных блюд;
- специальные цели по калориям и макроэлементам на любой день недели;
- настройки калорий из упражнений;
- приоритетная служба поддержки;
- отсутствие рекламы.

### 1.1.3 Сравнительный анализ прочих систем

В ходе работы были проанализированы прочие существующие аналогичные системы контроля и планирования рациона питания и реализованные в них функции. Обзор функций данных систем представлен в таблице 5.

Таблица 5 – Функции аналогичных систем

Функция \ Система	LifeSum	XBody Build	Бутерброд	Yazio	Good Food
Расчет дневной нормы калорий (по 3 целям)	+	+	-	+	+
Учет потребления воды	+	+	-	+	+
Статистика	+	+	+	+	+
Дневная норма БЖУ, воды	+	+	-	+	-
Дневная норма нутриентов	В платной версии	-	-	-	-
Индивидуальный план питания	В платной версии	-	-	В платной версии	-
Интеграция с трекерами активности	+	-	-	-	-

Проведенный анализ существующих аналогов разрабатываемой системы позволил выявить преимущества и недостатки того или иного набора функций анализируемых систем, что позволяет определиться с набором функций, которые будут реализованы в разрабатываемой системе контроля и планирования рациона питания.



### 1.3 Определение функций системы

Изучение диетологических принципов и методов процесса ведения здорового питания, а также анализ функций существующих аналогов разрабатываемой системы позволило выделить набор функций, которыми должна обладать система. Данные функции позволят осуществлять контроль и планирование рациона питания, необходимые для поддержания здорового образа жизни, похудения или набора массы.

Функции, которые должны быть реализованы в системе контроля и планирования рациона питания:

- расчет рекомендуемой суточной нормы потребления калорий, белков, жиров и углеводов с учетом индивидуальных особенностей человека, а также его образа жизни;
- полная информация о продуктах и содержании в них различных жизненно важных нутриентов, витаминов;
- возможность поиска продукта по категории или наименованию;
- расчет рекомендуемой нормы потребления питьевой воды;
- учет потребления питьевой воды, пищи и нутриентов;
- отслеживание изменений веса и замеров пользователя;
- возможность просматривать информацию о продуктах не осуществляя вход в систему.

#### **Вывод по первому разделу**

Изучение основных методов и принципов диетологии позволило выявить основные аспекты контроля и планирования рациона питания, а также выбрать метод расчета рекомендуемой суточной нормы потребления калорий.

Сравнительный анализ существующих систем контроля и планирования питания позволил выявить преимущества и недостатки этих систем, что позволяет определить функции, которые должны быть присущи системе контроля и планирования рациона питания.

## 2 Моделирование системы

### 2.1 Обоснование выбора базы данных продуктов питания

Одним из наиболее важных этапов в создании автоматизированной системы контроля и планирования рациона питания был подбор подходящей базы данных продуктов питания. Собственноручное формирование базы данных продуктов питания является крайне трудоемким процессом, поэтому было принято решение о поиске соответствующей базы в открытом доступе.

Было решено, что база данных, подходящая для решения поставленных задач, должна удовлетворять следующим критериям:

- доступность (возможность беспрепятственной реализации, в том числе хранение, изменение, копирование информации, а также нахождение её в открытом доступе);
- наличие деления на категории;
- наличие русскоязычного справочника;
- наличие информации о жизненно необходимых пищевых компонентах (нутриентах).

Результатом поиска стала база данных продуктов питания (USDA), соответствующая всем указанным критериям.

База данных продуктов питания (USDA) — национальная база данных продуктов питания, созданная Министерством сельского хозяйства США, содержащая данные о составе и энергетической ценности продуктов питания [31].

База данных содержит несколько разделов: описание пищи, пищевой ценности, вес, размеры, примечания, источники данных. Всего 25 пищевых групп, в числе которых измеряются: пищевая ценность, липиды, влажность, углеводы, энергия, минералы, аскорбиновая кислота, тиамин, определённые

витамины, протеин, фолиевая кислота, холин, жирные кислоты, холестерин, аминокислоты, вес и размеры и прочее.

Так как выбранная база данных сформирована из различных источников на территории США и предоставлена на английском языке, было принято решение о её локализации. В том числе, в базе данных продуктов питания (USDA) содержится информация о некоторых торговых марках, не реализуемых на территории РФ. Данная информация не актуальна для российских пользователей, которые являются целевой аудиторией сайта.

## 2.2 Обоснование выбора программных средств

Для разработки системы был выбран наиболее распространенный набор технологий для создания динамических сайтов: HTML, CSS, JavaScript, PHP, MySQL [5].

HTML – это стандартизированный язык гипертекстовой разметки документов, предназначенный для структурирования и представления содержимого сайта [11].

CSS – это формальный язык описания внешнего вида документа, написанного с использованием языка разметки. Используется как средство оформления внешнего вида веб-страниц, написанных с помощью html [10].

JavaScript – это язык программирования, который будет использоваться, как язык сценариев для придания интерактивности веб-страницам [3].

PHP – скриптовый язык общего назначения, интенсивно применяемый для разработки веб-приложений [22].

MySQL – свободная реляционная система управления базами данных. MySQL является расширением для малых и средних приложений [1]. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако в дистрибутив входит библиотека

внутреннего сервера, позволяющая включать MySQL в автономные программы [23].

В качестве средства для визуального проектирования базы данных был выбран инструмент – MySQL Workbench. Данный программный продукт позволяет наглядно представить модель базы данных в графическом виде, а затем сгенерировать SQL-скрипт, который впоследствии можно использовать для создания и эксплуатации базы в любой системе [2].

Для более эффективного процесса разработки было принято решение об использовании фреймворка, реализующего парадигму MVC.

Фреймворк – заготовки, шаблоны для программной платформы, определяющие архитектуру программной системы; программное обеспечение, облегчающее разработку и объединение разных модулей программного проекта [4].

MVC – схема разделения данных приложения, пользовательского интерфейса и управляющей логики на три отдельных компонента: модель, представление и контроллер [12].

На данный момент существует множество различных фреймворков на языке PHP. Проанализировав их возможности и особенности, было принято решение об использовании объектно-ориентированного компонентного фреймворка, реализующего парадигму MVC – Yii2. Yii2 обладает высокой производительностью, относительно других фреймворков; генерирует базовый PHP-код для четырех базовых операций, используемых при работе с базами данных; имеет возможность подключения сторонних библиотек и так далее [21].

Для ускорения процесса создания веб-интерфейса возникла необходимость в использовании bootstrap. Создание адаптивного дизайна является крайне трудоемким процессом без использования сетки bootstrap [6]. Yii2 поддерживает данный набор инструментов, что являлось одним из наиболее важных критериев выбора фреймворка.

В качестве средства для администрирования СУБД MySQL был выбран веб-интерфейс – phpMyAdmin. Данное веб-приложение позволяет администрировать сервера MySQL, запускать SQL-команды, а также просматривать содержимое баз данных и таблиц. Интерфейс phpMyAdmin крайне удобен в использовании и позволяет управлять базой данных с минимальным использованием скриптов SQL [13].

Для тестирования системы контроля и планирования рациона питания в процессе разработки возникла необходимость в наличии локального веб-сервера. Выбор пал на портативный локальный сервер OpenServer. Данный инструмент имеет большой набор встроенных инструментов, которые автоматически обновляются до актуальных версий. Важным преимуществом данного веб-сервера является возможность настройки HTTP, СУБД и PHP модулей в нужной комбинации в визуальном интерфейсе приложения.

В качестве редактора кода была выбрана коммерческая кросс-платформенная интегрированная среда разработки для PHP – PhpStorm. Данный программный продукт представляет собой интеллектуальный редактор для PHP, HTML и JavaScript с возможностями непрерывного анализа кода, предотвращение ошибок, автоматизированными средствами рефакторинга [18].

Возможности PhpStorm, определившие выбор данной интегрированной среды разработки:

- «умное» автодополнение кода;
- детектор дублируемого кода;
- MVC представление для Yii;
- PHP Code Sniffer, непрерывно проверяющий код в процессе разработки;
- поддержка SQL и баз данных;
- легко конфигурируемый визуальный отладчик;
- наличие плагинов, ускоряющих написание кода.

## 2.3 Проектирование базы данных

Для долгосрочного хранения информации о продуктах и о пользователях возникла необходимость в разработке базы данных. На данный момент существует множество систем управления базами данных, таких как MySQL, MariaDB, MongoDB, PostgreSQL и так далее [14]. В рамках данной работы предпочтение было отдано MySQL, так как в процессе обучения данная система управления базами данных была неоднократно использована, и она удовлетворяет всем критериям, таким как поддержка большого количества типов таблиц, полнотекстового поиска, открытая архитектура, коннектор для языка программирования PHP, кэширование запросов [9].

Сформулируем требования к базе данных:

- в базе данных должна храниться информация о пользователе, позволяющая однозначно идентифицировать его;
- должна храниться информация о личных данных пользователя, таких как вес, пол, рост, возраст и так далее;
- необходимо хранить данные продуктов, полученные из базы данных, сформированной на основе базы данных продуктов питания (USDA);
- должна храниться информация о категории продуктов питания для удобства использования системы;
- в базе данных необходимо хранить информацию об уровне активности пользователя и цели использования системы;
- должна храниться информация о содержании в продуктах питания нутриентов и витаминов в расчете на 100 грамм продукта;
- необходимо хранить информацию о том, когда, сколько и какие продукты были употреблены пользователем;
- необходимо хранить информацию о рекомендуемой суточной норме потребления пользователя.

С учетом требований, описанных выше, была спроектирована база данных средствами MySQL Workbench, состоящая из следующих сущностей:

- **UserData** – данная сущность используется для хранения персональных данных пользователя, таких как рост, пол, возраст, индекс массы тела, необходимых для расчета рекомендуемой нормы потребления и связана с сущностями **Activity**, **Goal**, **User** (отношение один к одному), отношением многие ко многим связана с **Measure** при помощи связующей сущности **UserData\_has\_Measure** и связью один ко многим с **UserWeight**;

- **User** – данная сущность предназначена для хранения данных для входа пользователей;

- **UserWeight** – предназначение данной сущности состоит в хранении данных о весе пользователя и дате измерения;

- **Measure** – сущность, используемая для хранения различных видов замеров, которые может осуществить пользователь системы;

- **UserData\_has\_Measure** – данная сущность является связующей для **UserData** и **Measure** и предназначена для хранения значений замеров пользователей и даты;

- **IntakeInformation** – так как пользователь должен иметь возможность просмотреть потребленные продукты, возникла необходимость в формировании сущности, которая будет хранить вес продукта, дату употребления и будет связана с сущностью пользователя и той, в которой будет храниться список продуктов. Данная сущность реализует описанные требования;

- **Activity** – данная сущность предназначена для хранения списка уровней активности, которых может придерживаться пользователь системы. Эта информация необходима для формирования рекомендуемой нормы потребления;

- **Goal** – данная сущность необходима для хранения списка целей, которых может придерживаться пользователь сайта.

- Vitamins – предназначение данной сущности состоит в хранении списка витаминов, содержащихся в продуктах из списка сущности FoodInformation;
- CategoryOfFood – каждый продукт в базе данных должен быть отнесен в некую категорию, данная сущность будет хранить в себе набор доступных категорий и будет связана с таблицей, хранящей информацию о продуктах питания (отношение один ко многим);
- FoodInformation – данная сущность будет хранить в себе список продуктов питания, связана с сущностями Vitamins, MacroMicroElements, Base (отношение многие ко многим);
- FoodInformation\_has\_Vitamins – это связующая сущность, позволяющая реализовать отношение многие ко многим между сущностью FoodInformation и Vitamins. Также данная сущность хранит в себе значение содержания конкретного витамина из списка в расчете на 100 грамм определенного продукта;
- Base – данная сущность предназначена для хранения списка базовых нутриентов, содержащихся в продуктах питания, что обуславливает необходимость реализации связи многие ко многим с сущностью FoodInformation;
- FoodInformation\_has\_Base – связующая сущность, позволяющая реализовать отношение многие ко многим между FoodInformation и Base;
- MacroMicroElements – аналогично сущностям Vitamins и Base, данная сущность хранит список макро- и микроэлементов, содержащихся в продуктах питания;
- MacroMicroElements\_has\_FoodInformation – аналогично сущностям FoodInformation\_has\_Base и FoodInformation\_has\_Vitamins является связующей сущностью, позволяющей реализовать отношение многие ко многим.



Спроектированные сущности со списком и кратким описанием атрибутов, включая первичные (pk – primary key) и внешние ключи (fk – foreign key), представлены в таблице 6.

Таблица 6 – Спроектированные сущности со списком атрибутов

Сущность	Атрибут	Тип
UserData	IdUserData (pk)	int
	Growth	int
	Age	int
	Sex	Varchar(1)
	Imt	int
	Goal_idGoal (fk)	int
	Activity_idActivity (fk)	int
	User_idUser (fk)	int
Activity	idActivity (pk)	int
	Name	Varchar(45)
Goal	idGoal (pk)	int
	Name	Varchar(45)
User	Username	Varchar(16)
	Email	Varchar(255)
	Password	Varchar(32)
	AuthKey	int
	idUser (pk)	int
UserWeight	idUserWeight (pk)	int
	Weight	int
	DateWeight	DateTime
	UserData_idUserData (fk)	int
Measure	idMeasure (pk)	int
	Description	Varchar(15)

Продолжение таблицы 6

Сущность	Атрибут	Тип
UserData_has_Measure	UserData_idUserData (fk)	int
	Measure_idMeasure (fk)	int
	Count	int
	MeasureData	DateTime
IntakeInformation	idIntake (pk)	int
	Weight	int
	dateInfo	DateTime
	FoodInfo_idFood (fk)	int
	UserData_idUserData (fk)	int
FoodInformation	idFood (pk)	int
	foodName	Varchar(70)
	CategoryOfFood_idCategory(fk)	int
CategoryOfFood	idCategory (pk)	int
	categoryName	Varchar(85)
Vitamins	idVitamin (pk)	int
	Name	Varchar(15)
Base	idBase (pk)	int
	Name	Varchar(60)
MacroMicroElements	idElements (pk)	int
	elementName	Varchar(45)
FoodInformation_has_base	FoodInfo_idFood (fk)	int
	Count	int
	Base_idBase (fk)	int
FoodInformation_has_vitamins	FoodInfo_idFood (fk)	int
	Vitamins_idVitamin (fk)	int
	Count	int

На основе спроектированных сущностей и атрибутов была разработана даталогическая модель базы данных, отражающая связи между сущностями, атрибуты и их типы. Часть разработанной даталогической модели представлена на рисунке 1. Остальная часть даталогической модели БД представлена на рисунке А.1.

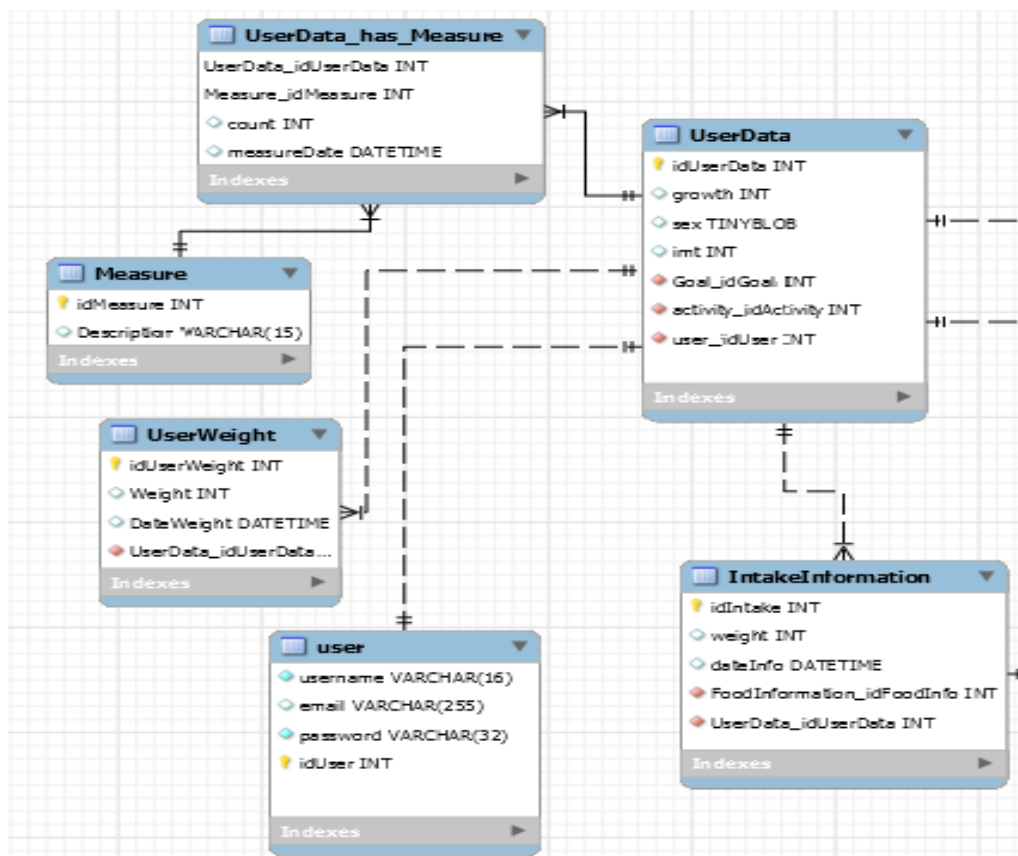


Рисунок 1 – Даталогическая модель базы данных

### Вывод по второму разделу

Выбор базы данных продуктов питания и программных продуктов позволил приступить к следующему этапу работы. Проведенное проектирование позволило минимизировать вероятность ошибок на этапе разработки базы данных и непосредственно системы.

## 3 Реализация системы

### 3.1 Разработка базы данных

Для того чтобы обеспечить пользователя возможностями контроля и планирования рациона питания, возникла необходимость о средстве отслеживания изменений, который пользователь вносит при работе с системой. Наилучшим решением для решения данной задачи является реляционная база данных со структурированной системой запросов [17].

База данных (БД) – совокупность организованной информации, относящейся к определенной предметной области, предназначенная для длительного хранения во внешней памяти компьютера и постоянного применения [19].

На основе спроектированной даталогической модели базы данных средствами MySQLWorkbench был сгенерирован SQL-скрипт, позволяющий создать базу данных, соответствующую ER-модели [24].

К спроектированной базе данных потребовалось добавить 5 таблиц, необходимых для реализации некоторых функций системы контроля и планирования рациона питания.

Таблица image предназначена для изменения размера, хранения, кэширования и прикрепления изображений к объектам сайта, в которых необходимо присутствие графической информации. Данная таблица хранит в себе следующую информацию:

- расположение файла в файловой системе;
- единый указатель ресурса(URL);
- имя модели, к элементу которой будет прикреплено данное изображение;
- уникальный идентификатор элемента и его изображения.

Таблица image и её атрибуты представлены в таблице 7.

Таблица 7 – Таблица image

Сущность	Атрибут	Тип
image	id	int(11)
	filePath	varchar(400)
	itemId	int(11)
	isMain	tinyint(1)
	modelName	varchar(150)
	urlAlias	varchar(400)
	name	varchar(80)

Таблица cb\_chart предназначена для хранения информации о диаграмме. В данной таблице хранится:

- уникальный идентификатор диаграммы;
- имя диаграммы;
- описание;
- тип диаграммы;
- уникальный идентификатор источника данных;
- тип визуализации;
- дата и время создания и изменения;
- уникальный идентификатор создавшего пользователя;
- информация о запросе;
- подписи осей;
- заголовок;
- подзаголовок;
- уникальный идентификатор подключения;
- наименования полей БД, отраженных в диаграмме;
- дополнительные параметры диаграммы.

Таблица cb\_chart и её атрибуты представлены в таблице 8.

Таблица 8 – Таблица cb\_chart

Сущность	Атрибут	Тип
cb_chart	id	varchar(36)
	name	varchar(255)
	detail	text
	chart_type	varchar(100)
	datasource_id	varchar(36)
	datasource_type	enum()
	tag_name	varchar(255)
	display_type	enum()
	result_id	varchar(36)
	created_at	datetime
	updated_at	datetime
	created_by	int(11)
	updated_by	int(11)
	query	text
	xaxis	varchar(100)
	xaxis_label	varchar(255)
	series	varchar(100)
	yaxis_label	varchar(255)
	title	varchar(255)
	sub_title	varchar(255)
	connection_id	varchar(36)
	options	text
	result	float
target_value	float	
stacked	smallint(1)	

Таблица `cb_chart_type` предназначена для хранения информации о доступных типах диаграмм, таких как круговая, линейчатая, график и другие.

Таблица `cb_chart_type` и её атрибуты отображены в таблице 9.

Таблица 9 – Таблица `cb_chart`

Сущность	Атрибут	Тип
cb_chart_type	code	varchar(100)
	name	varchar(255)
	image	varchar(255)
	options	text
	widget_classname	varchar(255)
	sort	int(11)

Таблица `cb_connection` предназначена для хранения информации о соединении с БД.

Таблица `cb_connection` и её атрибуты отображены в таблице 10.

Таблица 10 – Таблица `cb_connection`

Сущность	Атрибут	Тип
cb_connection	id	varchar(36)
	connection_name	varchar(255)
	host	varchar(30)
	port	varchar(10)
	username	varchar(150)
	password	varchar(150)
	driver	varchar(150)
	database	varchar(150)
	charset	varchar(150)
	created_at	datetime

Таблица `cb_datasource` предназначена для хранения информации об источнике данных.

Таблица `cb_datasource` и её атрибуты отображены в таблице 11.

Таблица 11 – Таблица `cb_datasource`

Сущность	Атрибут	Тип
cb_datasource	id	varchar(36)
	name	varchar(255)
	query	text
	connection_id	varchar(36)
	created_at	datetime
	updated_at	datetime

Результат создания БД, отраженный в веб-интерфейсе `phpMyAdmin`, представлен на рисунке 2. Подключение к БД представлено в листинге В.1.

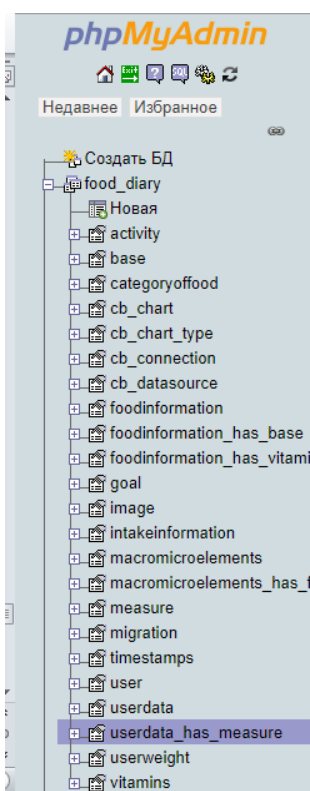


Рисунок 2 – Результат создания базы данных



### 3.2 Разработка интерфейса системы

Пользовательский интерфейс, или интерфейс – способ взаимодействия с устройством [25].

Современные средства создания динамических сайтов позволяют организовать удобный, интуитивно-понятный интерфейс. При разработке интерфейса следует учитывать корректность отображения с точки зрения сочетания разных цветов, иначе информация, представленная на сайте, станет нечитаемой [20].

Качественный web-интерфейс является не менее значимым фактором для удобства использования системы, чем её набор функций.

В ходе разработки интерфейса системы контроля и планирования рациона питания было принято решение о создании максимально простого и понятного способа взаимодействия с системой. Наличие многочисленных анимаций может повлечь за собой не только ухудшение быстродействия системы, но и привести к ухудшению восприятия важных функциональных аспектов. Конфигурация системы представлена в листинге В.2.

На начальном этапе разработки интерфейса системы контроля и планирования питания было разработано главное меню сайта, изображенное на рисунке 3.



Рисунок 3 – Главное меню сайта

Главное меню предоставляет доступ к основным элементам системы, таким как главная страница, категории продуктов, профиль пользователя, вход и регистрация.

Главная страница сайта должна четко отражать его суть, иметь навигационные элементы для удобного перехода между страницами, должна иметь приятный дизайн для того, чтобы заинтересовать пользователя.

Разработанная главная страница сайта представлена на рисунке 4.



Рисунок 4 – Главная страница

Следующим этапом было создание интерфейса для вывода категорий продуктов. Данная страница должна включать название категории, изображение, отражающее то, какие продукты входят в данную категорию и её краткое описание.

Результат создания данной страницы отображен на рисунке 5.

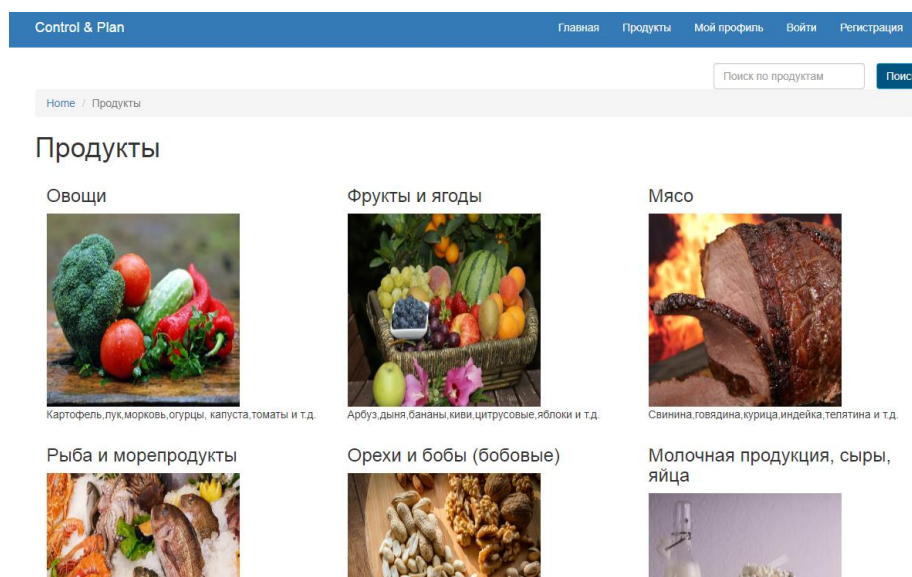


Рисунок 5 – Категории продуктов

Следующим этапом была разработана страница, отражающая список продуктов конкретной категории, представленная на рисунке 6.

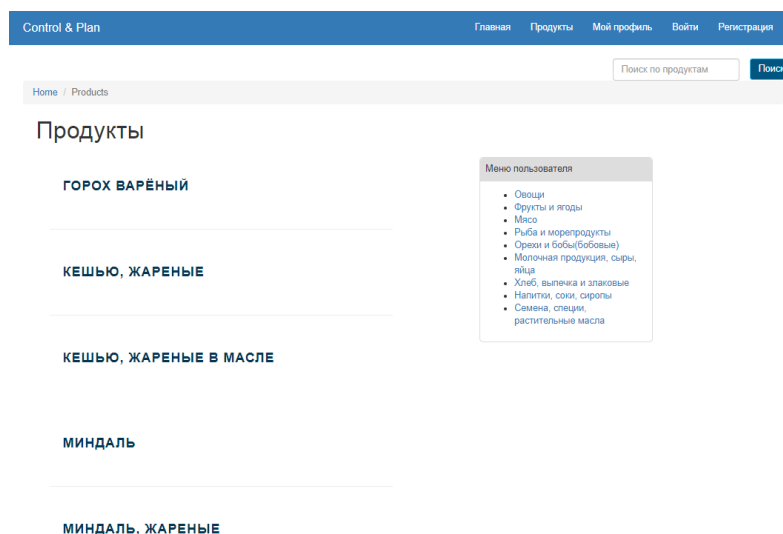


Рисунок 6 – Список продуктов конкретной категории

Также была разработана страница, отражающая детальную информацию о продукте питания. Данная страница включает в себя 3 таблицы, на которых изображено содержание нутриентов в расчете на 100 грамм продукта: таблица базовых нутриентов, витаминов и макро-микроэлементов. Данная страница представлена на рисунке 7.

Control & Plan Главная   Продукты   Мой профиль   Войти   Регистрация

Поиск по продуктам

Home / Product

### Кешью, жареные:

Содержание нутриентов на 100 гр. продукта:

Нутриент	Количество
Вода	2.8
Ккал	607
Белки	21.23
Жиры	55.17
Углеводы	17.68
Пищевые волокна	10.5
Сахар	4.55
Холестерин	0

Витамин	Количество
A	1
C	0.5
B6	0.118
B12	0
E	25.97
D	0
K	0

Макро/Микроэлемент	Количество
Кальций	291
Железо	3.68
Магний	274
Фосфор	466
Калий	699
Натрий	339
Цинк	3.07
Медь	0.955
Марганец	2.46
Селен	4.1
Тиамин	0.092
Рибофлавин	0.781
Ниацин	3.665
Пантотеновая кислота	0.229
Фолаты	27
Фолиевая кислота	0
Холин	52.1
Ретинол	0
Альфа-каротин	0
Бета-каротин	1
Бета-криптоксантин	0
Ликопин	0
Лютеин+зеаксантин	1
Насыщенные жирные кислоты	4.208
Мононенасыщенные жирные кислоты	34.793

Рисунок 7 – Подробная информация о продукте питания

Остальные страницы сайта будут представлены в разделе тестирования системы. Разработанный интерфейс системы контроля и планирования питания является интуитивно понятным, удобным и простым в использовании. Также важным преимуществом разработанного интерфейса системы является его адаптивность. Адаптивный дизайн позволяет отображать содержимое веб-сайта корректно на устройствах с различными размерами дисплея.

### 3.3 Программная реализация системы

После выполнения предыдущих этапов выпускной квалификационной работы появилась возможность приступить к реализации программной части системы.

Выбранный фреймворк yii2 обладает многочисленными возможностями для оптимизации процесса разработки. Одной из таких возможностей является наличие Gii (автоматический генератор кода), реализующий некоторые общие функции веб-сайта.

Gii позволяет сгенерировать модели, контроллеры, формы, модули и расширения для системы, а также четыре базовые функции для работы с базой данных (создание, чтение, обновление, удаление).

Система контроля и планирования питания разрабатывалась на основе архитектурного паттерна MVC (Model, View, Controller). Данный шаблон проектирования предполагает разделение данных приложения, пользовательского интерфейса и управляющей логики на три отдельных компонента [35].

Модель предоставляет данные и реагирует на команды контроллера, изменяя своё состояние. Пример кода модели представлен в листинге В.4.

Контроллер интерпретирует действия пользователя, оповещая модель о необходимости изменений.

Представление отвечает за отображение данных модели пользователю, реагирую на изменения модели [34].

Основная часть моделей и контроллеров, связанные с взаимодействием с сущностями базы данных, были созданы при помощи Gii. Впоследствии в них были добавлены различные методы, необходимые для реализации всех функций системы. Также возникла необходимость в создании некоторых моделей и контроллеров, не реализующих взаимосвязь с базой данных, но имеющих большое значение для создания программной части системы контроля и планирования питания.

Для интерпретации действий пользователя и оповещения моделей о необходимости изменений было создано 18 контроллеров. Список созданных в системе контроллеров изображен на рисунке 8.

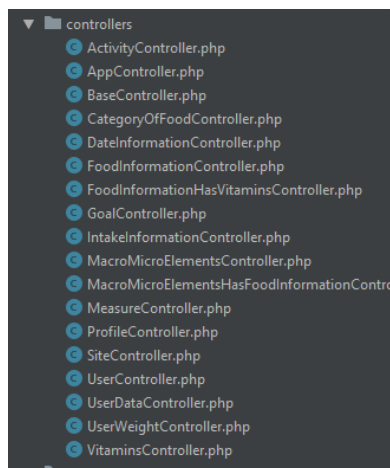


Рисунок 8 – Список контроллеров

Для реализации функций системы были разработаны более 30 моделей. Модель предоставляет данные и реагирует на команды контроллера [33]. Список созданных в системе моделей представлен на рисунке 9.

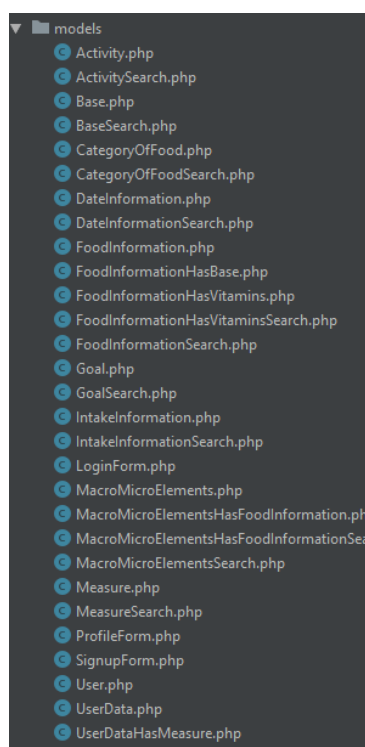


Рисунок 9 – Список моделей

Посредством Gii были сформированы представления для таблиц из базы данных, включающие в себя 5 файлов представлений. Список представлений для одной из таблиц представлен на рисунке 10.

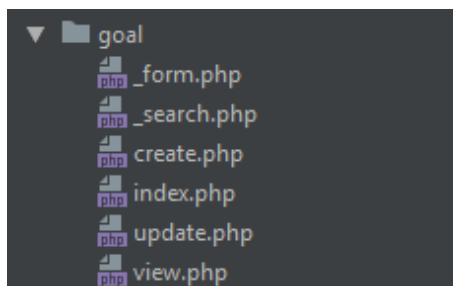


Рисунок 10 – Список представлений для таблицы Goal

Файл index.php является главной страницей для одной из сущностей БД, предоставляя в виде таблицы информацию обо всех её элементах. Пример одного из представлений index.php изображен на рисунке 11.

## Goals

Create Goal

Showing 1-3 of 3 items.

#	Name	Id Goal	
	<input type="text"/>	<input type="text"/>	
1	Похудеть	1	  
2	Сохранить вес	2	  
3	Набрать	3	  

Рисунок 11 – Данные таблицы goals в представлении index

Представление update.php предназначено для визуализации процесса изменения данных таблицы. Результат работы файла update.php для таблицы Goal представлен на рисунке 12.

## Update Goal: Похудеть

Name

Save

Рисунок 12 – Изменение элемента таблицы goals

Остальные сгенерированные представления для взаимодействия с таблицами базы данных выглядят схожим образом.

Для реализации возможности изменения размера, хранения, кэширования и прикрепления изображений к объектам сайта было использовано расширение yii2-images. Данное расширение позволяет оптимизировать работу системы с изображениями, ускорить процесс загрузки изображений на сайт, благодаря кэшированию [32].

Установленное расширение в папке проекта изображено на рисунке 13.

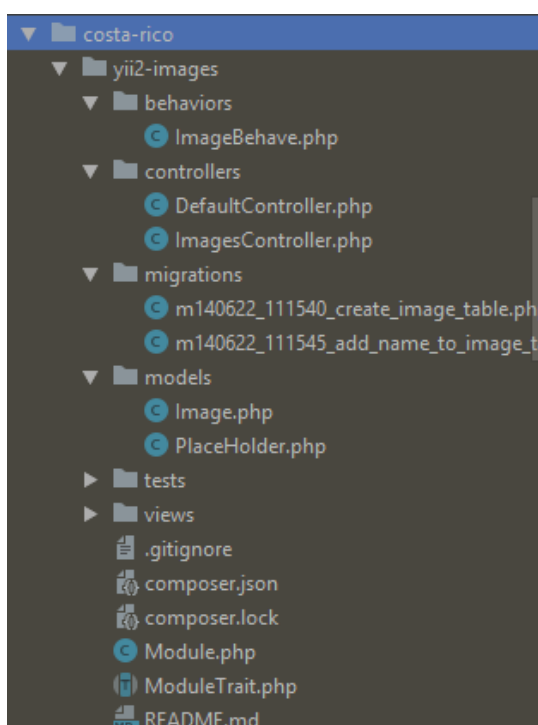


Рисунок 13 – Расширение Yii2-images

Для отображения прогресса пользователя по некоторым показателям, таким как вес или замеры тела, было принято решение о выводе информации в виде графиков и диаграмм. Для реализации данной функции системы контроля и планирования питания было установлено расширение chart-builder, позволяющее строить графики и диаграммы, на основе имеющейся в базе данных информации. Для построения графиков необходимо указать настройки подключения к БД, а также источник данных, в качестве которого может выступать запрос [28].



На рисунке 14 изображены типы диаграмм, имеющиеся в данном расширении, такие как график, круговая диаграмма, диаграмма области и другие.

### Chart Types

Charts Datasources Connections **Chart Type**

[+ Create Chart Type](#)

Showing 1-7 of 7 items.

#	Code	Name	
	<input type="text"/>	<input type="text"/>	
1	Area	Area	
2	Bar	Bar	
3	Column	Column	
4	Line	Line	
5	Pie	Pie	
6	Scatter	Scatter	
7	SolidGauge	Solid Gauge	

Рисунок 14 – Таблица с типами диаграмм

На рисунке 15 изображено создание подключения к базе данных для реализации диаграмм, связанных со значениями, хранящимися в той или иной таблице БД. Для создания подключения необходимо указать SQL драйвер, имя БД, пароль для доступа к БД, хост, порт, имя подключения, имя пользователя и кодировку.

### testConnection

Connection Name <input type="text" value="testConnection"/>	Driver <input type="text" value="Mysql"/>
Host <input type="text" value="127.0.0.1"/>	Charset <input type="text" value="utf8"/>
Port <input type="text" value="3306"/>	Database <input type="text" value="food_diary"/>
Username <input type="text" value="root"/>	Password <input type="text" value="****"/>

[Update](#)

Рисунок 15 – Создание подключения к базе данных

Источник данных позволяет отображать на любой из диаграмм выборку, взятую по запросу из данного источника [27]. На рисунке 16 представлено создание источника данных.

The screenshot shows a web interface for creating a data source. At the top, the title is 'Create Datasource'. Below it are several input fields: 'Name' (empty), 'Connection Id' (containing 'Default'), 'Params' (with a button 'เพิ่มส่วนปร...'), and a 'Query' editor area (with a '1' in the top left corner).

Рисунок 16 – Создание источника данных

На рисунке 17 представлено прикрепление источника данных, из которого будет происходить выборка на график или диаграмму через имеющийся источник данных или SQL-запрос.

The screenshot shows a web interface for attaching a data source. At the top, there are three tabs: 'Detail', 'Datasource', and 'Preview'. Below the tabs are several input fields: 'Datasource Type' (with radio buttons for 'Query' and 'Data Source'), 'Datasource ID' (containing 'userweight'), 'Connection Id' (containing 'testConnection'), 'Tag Name' (containing 'test'), and 'Params' (with a button 'เพิ่มส่วนปร...'). At the bottom, there are two buttons: 'Execute' and 'Update', and a date range '2019-01-01 - 2019-06-10'.

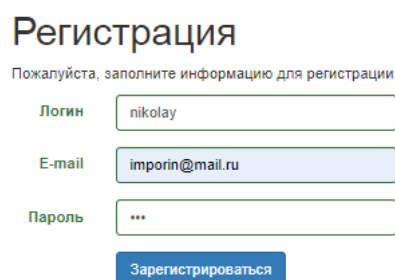
Рисунок 17 – Прикрепление источника данных

### 3.4 Тестирование и отладка системы

Тестирование и отладка работы системы является одним из наиболее важных этапов реализации, так как позволяет выявить неточности работы системы и проверить соответствие между реальным поведением и ожидаемым на конечном наборе тестов, выбранных определенным образом [26].

В первую очередь была протестирована форма регистрации пользователей.

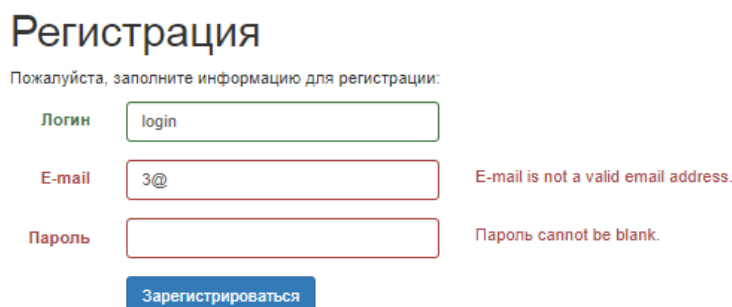
Разработанная форма регистрации содержит поля: логин, e-mail, пароль. На рисунке 18 представлена форма регистрации с введенными в нее корректными данными.



The screenshot shows a registration form titled "Регистрация". Below the title is the instruction "Пожалуйста, заполните информацию для регистрации:". There are three input fields: "Логин" with the value "nikolay", "E-mail" with the value "imporin@mail.ru", and "Пароль" with three asterisks. A blue button labeled "Зарегистрироваться" is positioned below the password field.

Рисунок 18 – Регистрация

Система должна реагировать на некорректный ввод данных, на рисунке 19 отображена реакция программы на ввод пустого пароля и некорректного e-mail адреса.



The screenshot shows the same registration form as in Figure 18, but with invalid data. The "Логин" field contains "login". The "E-mail" field contains "3@" and has a red border with the error message "E-mail is not a valid email address." to its right. The "Пароль" field is empty and has a red border with the error message "Пароль cannot be blank." to its right. The "Зарегистрироваться" button is still present.

Рисунок 19 – Ввод некорректных данных при регистрации

В результате регистрации пользователя в базе данных появляется соответствующая запись. Для обеспечения безопасности, пароль пользователя хранится в хэше, результат записи в БД представлен на рисунке 20.

idUser	email	password	username
19	imporin@mail.ru	\$2y\$13\$xt8TzwhgVy4eiqLx.La6OQlgdCs1cBvv0HAgtVG0O...	nikolay

Рисунок 20 – Результат регистрации в БД

Следующим этапом была протестирована возможность входа в систему. При закрытии браузера и повторном запуске системы, не будет происходить выхода из системы, если пользователь оставил активным пункт «Запомнить меня». Вход в систему с именем пользователя nikolay представлен на рисунке 21.

Home / Вход

### Вход

Пожалуйста, заполните информацию для входа в систему:

Логин

Пароль

Запомнить меня

Рисунок 21 – Вход в систему

Активная опция «Запомнить меня» реализует возможность кэширования данных входа. Информация для обеспечения кэширования хранится в БД в поле auth\_key. Результат записи в поле auth\_key изображен на рисунке 22.

idUser	email	password	username	auth_key
19	imporin@mail.ru	\$2y\$13\$xt8Tzv	nikolay	FDyaWOfbxc

Рисунок 22 – Результат сохранения кэша

В результате входа в систему появляется возможность использования функций системы, доступ к которым есть только у зарегистрированных пользователей. Результат входа в систему представлен на рисунке 23.

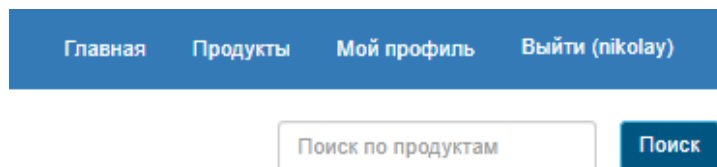


Рисунок 23 – Результат входа в систему

Следующим этапом был протестирован поиск по продуктам. При вводе информации в строку поиска и нажатии на кнопку «Поиск», должен происходить вывод всех продуктов питания, в которых встречается искомое слово. Результат поиска представлен на рисунке 24.

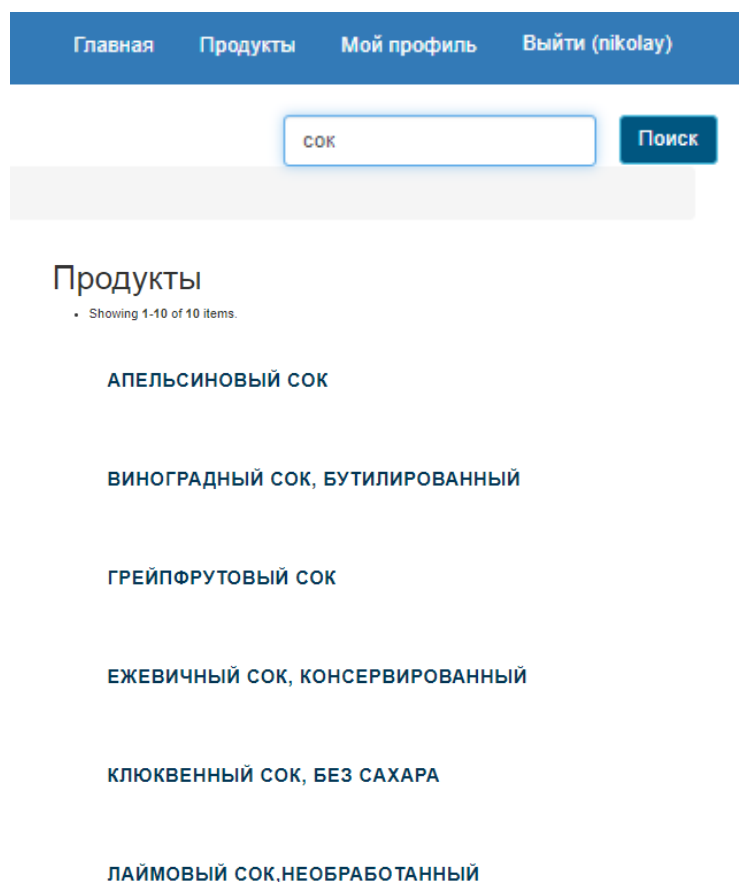


Рисунок 24 – Результат поиска

При нажатии на наименование продукта в результате поиска происходит переадресация на страницу с подробной информацией о данном продукте, аналогично переходу из списка продуктов по категориям. Результат перехода на страницу продукта изображен на рисунке 25.

Control & Plan Главная | Продукты | Мой профиль | Выйти (nikolay)

Поиск по продуктам

Home / Product

### Грейпфрутовый сок:

Содержание нутриентов на 100 гр. продукта:

Нутриент	Количество
Вода	90.1
Ккал	38
Белки	0.52
Жиры	0.1
Углеводы	8.96
Пищевые волокна	0.1
Сахар	8.86
Холестерин	0

Витамин	Количество
A	7
C	38
B6	0.02
B12	0
E	0.04
D	0
K	0

Макро/Микроэлемент	Количество
Кальций	7
Железо	0.2
Магний	10
Фосфор	11
Калий	163
Натрий	1
Цинк	0.09
Медь	0.038
Марганец	0.02
Селен	0.1
Тиамин	0.042
Рибофлавин	0.02
Ниацин	0.231
Пантотеновая кислота	0.13

Рисунок 25 – Переход на страницу продукта

На главной странице было организовано дополнительное меню, по которому можно осуществить переход к трём наиболее важным страницам сайта: продукты, мой профиль, отчеты. Переход по кнопке «Продукты» переадресовывает пользователя на страницу с категориями продуктов питания. Переход на страницу «Профиль» осуществляется по одноименной кнопке, а кнопка «Отчеты» позволяет перейти к графикам и диаграммам пользователя. Дополнительное меню представлено на рисунке 26.

#### Продукты

Control & Plan предоставляет в свободном доступе информацию о многочисленных продуктах питания и напитках, их энергетической ценности, содержащихся в них витаминах, макро- и микроэлементах. База данных продуктов питания постоянно расширяется и обновляется. Для поиска конкретного продукта предусмотрен поиск по наименованию.

#### Мой профиль

Зарегистрированные пользователи могут отслеживать свои результаты, вносить информацию о том, когда, сколько и какие продукты были употреблены, контролировать объем употребленной воды, записывать информацию об изменениях собственного веса и различных замеров объемов тела.

#### Отчеты

Control & Plan позволяет не только записывать результаты, но и отслеживать их в виде графиков и диаграмм. На данном сайте Вы можете отслеживать свои результаты в наиболее удобной для Вас манере. Данные функции доступны для зарегистрированных пользователей.

Рисунок 26 – Меню на главной странице

При переходе по ссылке «Продукты» из главного меню сайта осуществляется переадресация на страницу с категориями продуктов питания, данная страница представлена на рисунке 27.

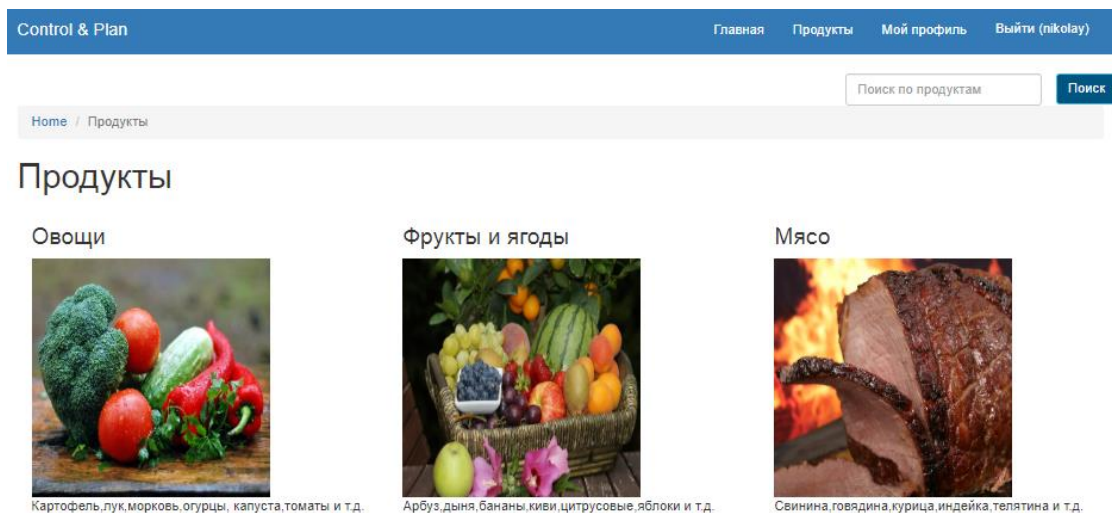


Рисунок 27 – Категории продуктов

При нажатии на изображение или наименование категории происходит переход ко всем продуктам данной категории. Список продуктов конкретной категории изображен на рисунке 28.

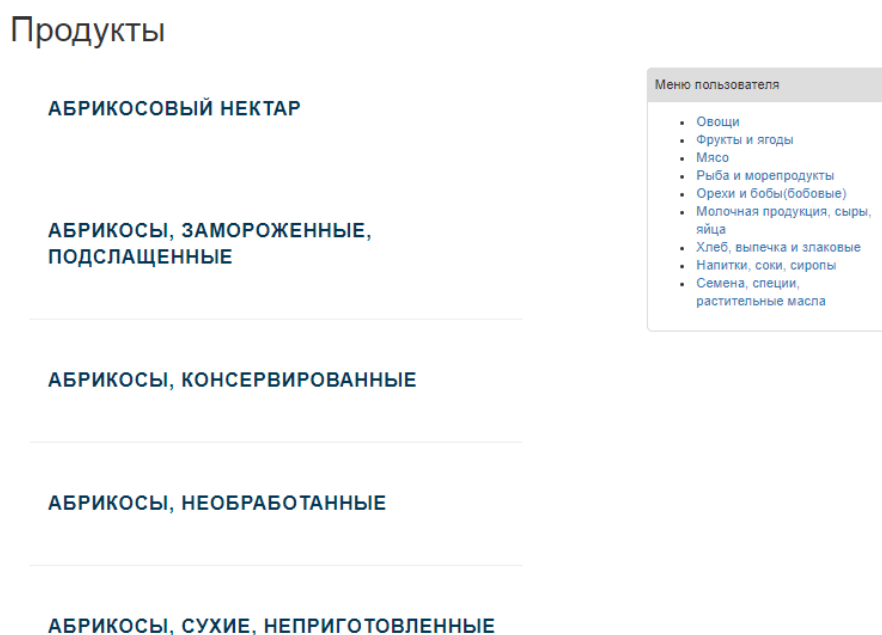


Рисунок 28 – Продукты категории «Фрукты и ягоды»

На странице, отображающей все продукты конкретной категории, было организовано дополнительное меню, позволяющее перейти к другой категории продуктов. Дополнительное меню и переход к другой категории представлены на рисунке 29.

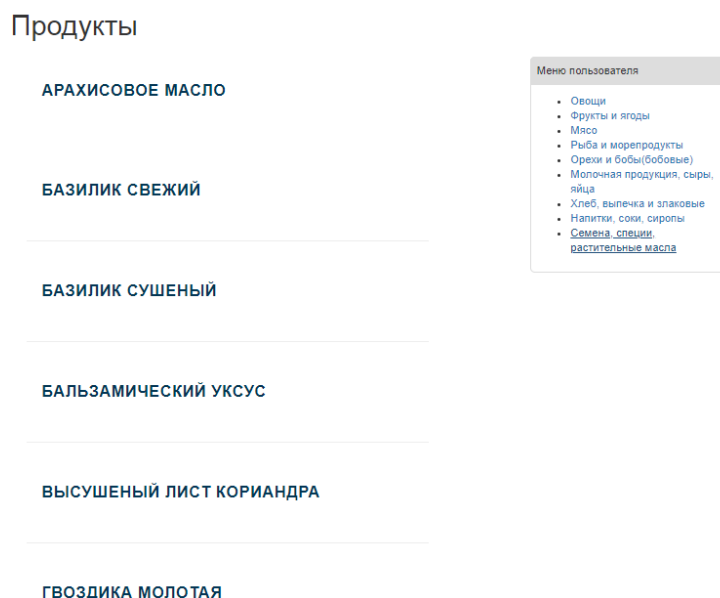


Рисунок 29 – Переход к другой категории из меню категорий

Для расчета рекомендуемой суточной нормы потребления пользователя необходима подробная информация о нем. Форма ввода данных пользователя представлена на рисунке 30.

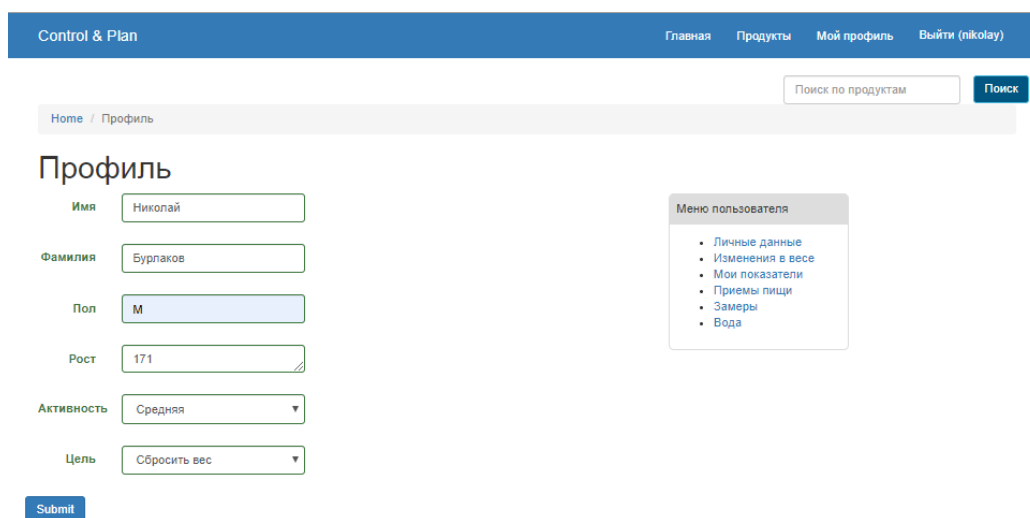


Рисунок 30 – Личные данные пользователя



Изменения в весе пользователь может вводить в любое удобное для него время и отслеживать на графике. Страница, отображающая изменения в весе пользователя и график изменений представлена на рисунке 31.

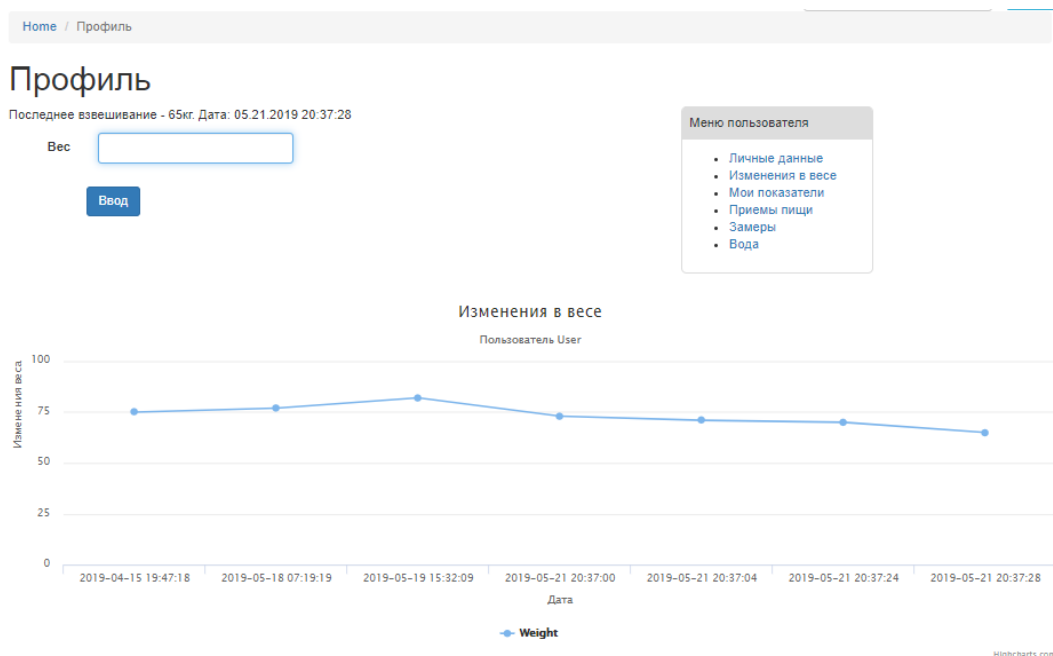


Рисунок 31 – Изменения в весе

Основой контроля питания является наличие информации о том, когда, сколько и какой пищи было употреблено, а также о её энергетической ценности. Данная страница представлена на рисунке 32.

## Приемы пищи

**Внести информацию**

Ваше рекомендуемое суточное количество (РСК) составляет 2750 к/кал.

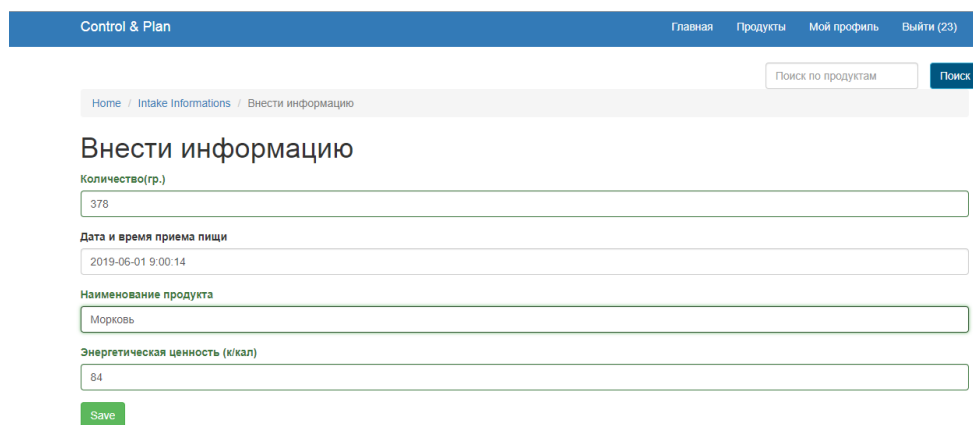
На данный момент употреблено 15% (413 к/кал)

Showing 1-3 of 3 items.

#	% От РСК	Количество(гр.)	Дата и время приема пищи	Наименование продукта	Энергетическая ценность (к/кал)	
	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
1	1.49	20	2019-05-09 17:05:14	Ржаной хлеб	41	
2	10.8	250	2019-05-09 08:30:28	Куриная грудка, жареная	297	
3	2.73	19	2019-05-09 21:07:01	Сыр пармезан	75	

Рисунок 32 – Приемы пищи

Для ввода новой информации об употреблении пищи была разработана форма, включающая в себя количество, дату и время приема, наименование продукта. На рисунке 33 представлен ввод новой информации об употребленной пище.



The screenshot shows a web application interface with a blue header bar containing the text "Control & Plan" and navigation links: "Главная", "Продукты", "Мой профиль", and "Выйти (23)". Below the header is a search bar with the text "Поиск по продуктам" and a "Поиск" button. The main content area has a breadcrumb trail: "Home / Intake Informations / Внести информацию". The title of the form is "Внести информацию". The form contains four input fields: "Количество(гр.)" with the value "378", "Дата и время приема пищи" with the value "2019-06-01 9:00:14", "Наименование продукта" with the value "Морковь", and "Энергетическая ценность (к/кал)" with the value "84". At the bottom of the form is a green "Save" button.

Рисунок 33 – Ввод информации об употребленной пище

### 3.5 Метод SWOT-анализа

Для осуществления начальной оценки текущей ситуации в сфере контроля и планирования рациона питания, а также сравнения системы, разрабатываемой в ходе выполнения выпускной квалификационной работы с существующими аналогами, было принято решение об использовании метода SWOT-анализа.

SWOT-анализ – метод стратегического планирования, заключающийся в выявлении внешних и внутренних факторов, способных описать разработанную систему и её аналоги. Факторы SWOT-анализа разделены на 4 категории: сильные стороны, слабые стороны, возможности, угрозы.

Задача SWOT-анализа — дать структурированное описание ситуации, относительно которой нужно принять какое-либо решение.

Был осуществлен SWOT-анализ разрабатываемой системы контроля и планирования рациона питания, позволивший выявить сильные и слабые стороны системы с точки зрения внешних и внутренних факторов,

способных повлиять на разработку. В результате SWOT-анализа была получена SWOT-матрица, представленная в таблице 12.

Таблица 12 – SWOT-матрица

Сильные стороны	Возможности	
	Расширение круга потребителей	Совершенствование разработки
Низкая стоимость разработки	++	0
Многофункциональность	++	+
Обеспечение сопровождения	+	++
Удобство использования	++	0
Масштабируемость	+	+
Доступ с любого устройства через интернет	++	+
Итого	+10	+5
Слабые стороны		
Узконаправленность	--	0
Несовместимость с другим программным обеспечением	-	0
Отсутствие опыта реализации крупных проектов	--	--
Невысокая репутация компании	--	0
Итого	-7	-2
Общий итог	+3	+3

Продолжение таблицы 12

Сильные стороны	Угрозы		Итого
	Появление новых конкурентов	Моральное устаревание	
Низкая стоимость разработки	++	0	+4
Многофункциональность	+	++	+6
Обеспечение сопровождения	+	+	+5
Удобство использования	+	0	+2
Масштабируемость	+	+	+4
Доступ с любого устройства через интернет	+	0	+4
Итого	+6	+4	+25
Слабые стороны			
Узконаправленность	-	-	-4
Несовместимость с другим программным обеспечением	-	-	-3
Отсутствие опыта реализации крупных проектов	-	0	-5
Невысокая репутация компании	--	0	-4
Итого	-5	-2	-16
Общий итог	+1	+2	+10

### **Вывод по третьему разделу**

В данном разделе выпускной квалификационной работы была осуществлена разработка базы данных, веб-интерфейса, программной части программы, произведено тестирование и отладка системы. Найдены и исправлены выявленные неточности в поведении программы, проведен SWOT-анализ, позволивший выявить сильные и слабые стороны разработанной системы.

## ЗАКЛЮЧЕНИЕ

В результате выполнения выпускной квалификационной работы была разработана системы контроля и планирования питания. Система отвечает поставленным требованиям, способна решать поставленные задачи, имеет удобный и интуитивно-понятный интерфейс.

В процессе выполнения выпускной квалификационной работы были решены следующие задачи:

- изучены и выбраны методы диетологии, используемые для расчета рекомендуемой суточной нормы потребления, что позволило выявить основные способы достижения контроля и планирования рациона питания;
- выбрана база данных продуктов питания, данные из которой были перенесены впоследствии в базу данных системы;
- проведен анализ существующих аналогов, позволяющий определить функции, которым должна обладать система;
- изучены и выбраны средства для разработки, позволяющие приступить к этапам проектирования и разработки системы;
- спроектирована система контроля и планирования питания, что позволило минимизировать ошибки на этапе разработки;
- разработана система контроля и планирования питания;
- проведено тестирование системы контроля и планирования питания, при котором были выявлены и исправлены некоторые ошибки в работе системы.

Разработанная система контроля и планирования рациона реализует следующие функции:

- расчет рекомендуемой суточной нормы потребления калорий, белков, жиров и углеводов с учетом индивидуальных особенностей человека, а также его образа жизни;

- полная информация о продуктах и содержании в них различных жизненно важных нутриентов, витаминов;
- возможность поиска продукта по его наименованию или категории;
- расчет рекомендуемой нормы потребления питьевой воды;
- учет потребления питьевой воды;
- учет потребления пищи;
- учет потребления нутриентов;
- отслеживание изменений веса и замеров пользователя;
- возможность просматривать информацию о продуктах не осуществляя вход в систему.

В результате была достигнута основная цель выпускной квалификационной работы: совершенствование процесса планирования и контроля питания человека посредством разработки системы.

В будущем, возможны улучшения разработанной системы в следующих направлениях:

- создание и реализация алгоритма автоматизированного подбора рациона питания;
- разработка мобильного приложения, реализующего прежние функции сайта и осуществляющего интеграцию с фитнес-трекерами активности;
- наличие информации о тренировках и о количестве потраченной энергии при различных видах нагрузки;
- учет различных дополнительных параметров при формировании рациона питания.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Бенедетти, Р. Изучаем работу с jQuery/Р. Бенедетти, Р. Крэнли. – СПб.: Питер, 2012. – 512с.
2. Бенкен, Е. С. PHP, MySQL, XML: Программирование для Интернета/Е. С. Бенкен. – СПб.: БХВ-Петербург, 2007. – 336с.
3. Бибо, Б. jQuery. Подробное руководство по продвинутому JavaScript/ Б. Бибо, И. Кац. – М.: Символ, 2012. – 384с.
4. Бьюли А. Изучаем SQL/А. Бьюли. – М.: Символ-плюс, 1986. – 721 с.
5. Вахтуров В. В. JavaScript. Освой на примерах/ В. В. Вахтуров. – М.: БХВ-Петербург, 2007. – 957 с.
6. Веллинг, Л. Разработка веб-приложений с помощью PHP и MySQL / Л. Веллинг, Л. Томсон. – М.: Вильямс, 2010. – 848с.
7. ГОСТ 7.1-2003. Библиографическая запись. Общие требования и правила составления. – Москва.: ИПК Изд-во стандартов, 2004. – 47с.
8. ГОСТ 7.82-2001. Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления. – Минск.: ИПК Изд-во стандартов, 2001. – 23 с.
9. Дамашке, Г. PHP и MySQL/ Г. Дамашке. – М.: НТ, 2016. – 569с.
10. Дронов, В. А. HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов/В. А. Дронов. – СПб.: БХВ-Петербург, 2011. – 416с.
11. Дронов, В. А. PHP, MySQL и Dreamweaver. Разработка интерактивных Web-сайтов/В.А. Дронов. – СПб.: БХВ-Петербург, 2007. – 480с.
12. Дунаев, В.В. HTML, скрипты и стили/ В. В. Дунаев. – М.: БХВ, 2006. – 832 с.
13. Закас, Н. JavaScript. Оптимизация производительности/Н. Закас. – СПб.: Символ, 2012. – 256с.

14. Зандстра, М. PHP. Объекты, шаблоны и методики программирования/М. Зандстра. – М.: Вильямс, 2016. – 560с.
15. Зинец, И. И. Здоровье и питание/И. И. Зинец. – Пермь: Алгос-Пресс, 1995. – 283с.
16. Кардаш, А. 100 самых полезных продуктов/ А. Кардаш. – М.: Вильямс, 2014. – 461с.
17. Колисниченко, Д. Н. PHP и MySQL. Разработка Web-приложений/ Д. Н. Колисниченко. – СПб.: БХВ-Петербург, 2017. – 560с.
18. Кригель А. SQL. Библия пользователя/А. Кригель. – М.: Вильямс, 2010. – 394с.
19. Кузнецов, М. В. PHP 5. Практика создания Web-сайтов/М. В. Кузнецов, И. В. Симдянов, С. В. Голышев. – СПб.: БХВ-Петербург, 2017. – 960с.
20. Лабберс, П. HTML5 для профессионалов: мощные инструменты для разработки современных веб-приложений/П. Лабберс. – М.: Вильямс, 2011. – 453с.
21. Лагутин, К.В. Пояснительная записка к курсовому проекту/ К.В. Лагутин. – М.: МГПУ, 2012. – 35с.
22. Ловэйн, П. PHP5. Объектно-ориентированное программирование/ П. Ловэйн. – М.: НТ Пресс, 2007. – 224с.
23. Локхарт, Д. Современный PHP. Новые возможности и передовой опыт/Д. Локхарт. – М.: ДМК Пресс, 2016. – 304с.
24. Ляпин, Д.А. PHP — это просто. Начинаем с видеоуроков (+ CD-ROM)/Д. А. Ляпин. – СПб.: БХВ-Петербург, 2017. – 881с.
25. Ляпин, Д. А. PHP – это просто. Начинаем с видеоуроков/Д. А. Ляпин. – СПб.: БХВ-Петербург, 2017. – 642с.
26. Макконелл С. Совершенный код. Мастер-класс/С. Макконелл. – М.: Русская редакция, 2016. – 896с.
27. Макфарланд, Д. Большая книга CSS3/Д. Макфарланд. – М.: Питер, 2015. – 608с.



28. Маккоу, А., Веб-приложения на JavaScript/ А. Маккоу. – М.: Питер, 2012. – 285с.
29. Михайлов, В. С. Растения на вашем столе/В. С. Михайлов, Л. А. Трушкина.– М.: Советская Россия, 2003. – 136с.
30. Рацион питания населения. 2013: Статистический сборник/Росстат-М.: ИИЦ «Статистика России», 2016 . – 220 с.
31. Скальный, А. В. Основы здорового питания: пособие по общей нутрициологии/А. В. Скальный, И. А. Рудаков, С. В. Нотова. – Оренбург.: ГОУ ОГУ, 2005. – 117 с.
32. Стеймец, У. РНР. 75 готовых решений для вашего сайта/ У. Стеймец. – СПб.: Наука и Техника, 2009. – 256 с.
33. Флэнаган, Д. JavaScript. Подробное руководство, 6-е издание/Д. Флэнаган. – М.: Символ-Плюс, 2012. – 1080с.
34. Хольцшлаг, М. Э. Использование HTML 4. Специальное издание (+ CD - ROM)/М. Э. Хольцшлаг. – М.: Вильямс, 2001. - 970 с.
35. Шлоснейгл, Д. Профессиональное программирование на РНР/Д. Шлоснейгл. – М.: Вильямс, 2006. – 345с.

# ПРИЛОЖЕНИЕ А

## ER-Модель БД

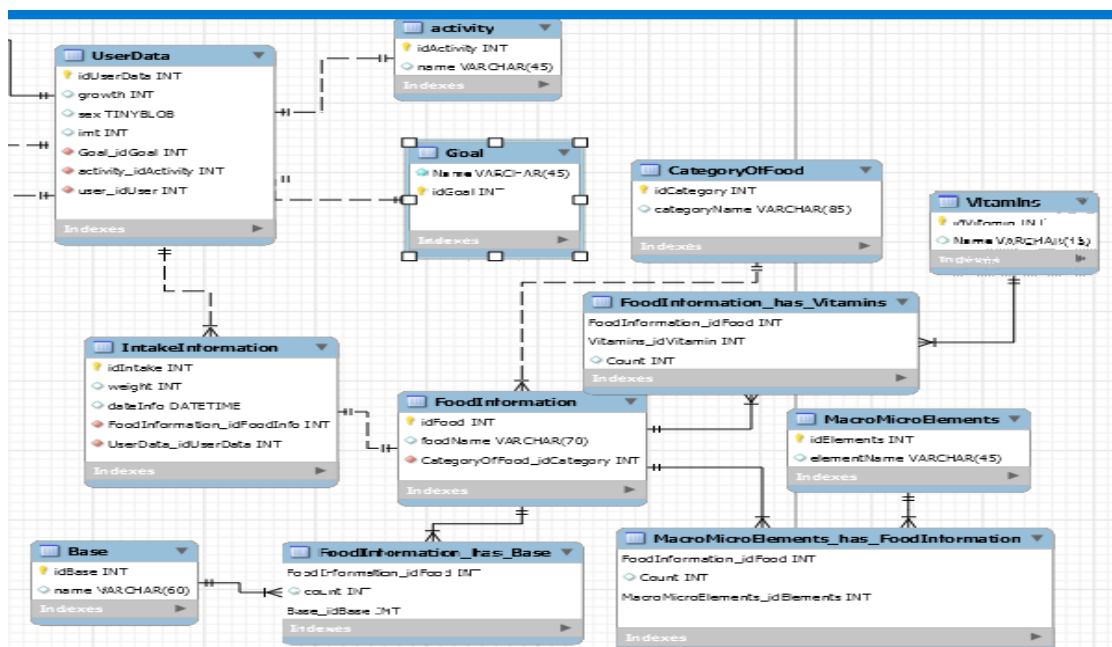


Рисунок А.1 – Даталогическая модель БД

## ПРИЛОЖЕНИЕ В

### Программный код

#### Листинг В.1 – Подключение к БД

```
<?php

return [
    'class' => 'yii\db\Connection',
    'dsn' => 'mysql:host=127.0.0.1;dbname=food_diary',
    'username' => 'root',
    'password' => 'root',
    'charset' => 'utf8',
];
```

#### Листинг В.2 – Конфигурация системы

```
<?php

$params = require __DIR__ . '/params.php';
$db = require __DIR__ . '/db.php';

$config = [
    'id' => 'basic',
    'basePath' => dirname(__DIR__),
    'bootstrap' => ['log'],
    'aliases' => [
        '@bower' => '@vendor/bower-asset',
        '@npm' => '@vendor/npm-asset',
    ],
    'modules' => [
        'chartbuilder'=>[
            'class'=> 'yii2learning\chartbuilder\Module'
        ],
        'gridview' => [
            'class' => '\kartik\grid\Module',
        ],
        'yii2images' => [
            'class' => 'rico\yii2images\Module',
```

```

//be sure, that permissions ok
//if you cant avoid permission errors you have to create "images" folder in web root manually and
set 777 permissions
    'imagesStorePath' => 'images/store/categories', //path to origin images
    'imagesCachePath' => 'images/cache', //path to resized copies
    'graphicsLibrary' => 'GD', //but really its better to use 'Imagick'
    'placeholderPath' => '@webroot/images/placeholder.png', // if you want to get placeholder when
image not exists, string will be processed by Yii::getAlias
    'imageCompressionQuality' => 100, // Optional. Default value is 85.
]
],
//'layout' => 'basic',
'components' => [
    'request' => [
        // !!! insert a secret key in the following (if it is empty) - this is required by cookie validation
        'cookieValidationKey' => 'U-9TB8Bz-67IKLNFaP-BlzKpU6rQvcOc',
        'baseUrl' => "",
    ],
    'cache' => [
        'class' => 'yii\caching\FileCache',
    ],
    'user' => [
        'identityClass' => 'app\models\User',
        'enableAutoLogin' => true,
    ],
    'errorHandler' => [
        'errorAction' => 'site/error',
    ],
    'mailer' => [
        'class' => 'yii\swiftmailer\Mailer',
        // send all mails to a file by default. You have to set
        // 'useFileTransport' to false and configure a transport
        // for the mailer to send real emails.
        'useFileTransport' => true,
    ],
    'log' => [
        'traceLevel' => YII_DEBUG ? 3 : 0,
        'targets' => [

```

```

        [
            'class' => 'yii\log\FileTarget',
            'levels' => ['error', 'warning'],
        ],
    ],
    'db' => $db,
    'urlManager' => [
        'enablePrettyUrl' => true,
        'showScriptName' => false,
        'rules' => [
            'products' => 'site/products',
            'product' => 'site/product',
            'profile' => 'profile/profile',
            'weight' => 'profile/weight',
            'intake' => 'profile/intake',
            'measure' => 'profile/measure',
            'water' => 'profile/water',
            'login' => 'site/login',
            'facts' => 'profile/facts',
            'categories' => 'category-of-food/categories'
        ],
    ],
    'params' => $params,
];

```

```

if (YII_ENV_DEV) {
    // configuration adjustments for 'dev' environment
    $config['bootstrap'][] = 'debug';
    $config['modules']['debug'] = [
        'class' => 'yii\debug\Module',
        // uncomment the following to add your IP if you are not connecting from localhost.
        //allowedIPs' => ['127.0.0.1', '::1'],
    ];

    $config['bootstrap'][] = 'gii';
    $config['modules']['gii'] = [

```

```

        'class' => 'yii\gii\Module',
        // uncomment the following to add your IP if you are not connecting from localhost.
        //'allowedIPs' => ['127.0.0.1', '::1'],
    ];
}

```

```
return $config;
```

### Листинг В.3 – Контроллер категорий продуктов

```
<?php
```

```
namespace app\controllers;
```

```

use app\models\FoodInformation;
use app\models\MacroMicroElementsHasFoodInformation;
use Yii;
use app\models\CategoryOfFood;
use app\models\CategoryOfFoodSearch;
use yii\data\ActiveDataProvider;
use yii\web\Controller;
use yii\web\NotFoundHttpException;
use yii\filters\VerbFilter;
use yii\db\ActiveQuery;
use yii\web\UploadedFile;

```

```
/**
```

```
 * CategoryOfFoodController implements the CRUD actions for CategoryOfFood model.
```

```
*/
```

```
class CategoryOfFoodController extends Controller
```

```
{
```

```
    /**
```

```
     * {@inheritdoc}
```

```
     */
```

```
    public function behaviors()
```

```
    {
```

```
        return [
```

```
            'verbs' => [
```

```
                'class' => VerbFilter::className(),
```

```
                'actions' => [
```

```

        'delete' => ['POST'],
    ],
],
];
}

/**
 * Lists all CategoryOfFood models.
 * @return mixed
 */
public function actionIndex()
{
    $searchModel = new CategoryOfFoodSearch();
    $dataProvider = $searchModel->search(Yii::$app->request->queryParams);

    return $this->render('index', [
        'searchModel' => $searchModel,
        'dataProvider' => $dataProvider,
    ]);
}

/**
 * Displays a single CategoryOfFood model.
 * @param integer $id
 * @return mixed
 * @throws NotFoundHttpException if the model cannot be found
 */
public function actionView($id)
{
    return $this->render('view', [
        'model' => $this->findModel($id),
    ]);
}

/**
 * Creates a new CategoryOfFood model.
 * If creation is successful, the browser will be redirected to the 'view' page.
 * @return mixed

```

```

*/
public function actionCreate()
{
    $model = new CategoryOfFood();

    if ($model->load(Yii::$app->request->post()) && $model->save()) {
        return $this->redirect(['view', 'id' => $model->idCategory]);
    }

    return $this->render('create', [
        'model' => $model,
    ]);
}

/**
 * Updates an existing CategoryOfFood model.
 * If update is successful, the browser will be redirected to the 'view' page.
 * @param integer $id
 * @return mixed
 * @throws NotFoundHttpException if the model cannot be found
 */
public function actionUpdate($id)
{
    $model = $this->findModel($id);

    if ($model->load(Yii::$app->request->post()) && $model->save()) {

        $model->image = UploadedFile::getInstance($model, 'image');
        if($model->image){

            $path = Yii::getAlias('@webroot/images/store/categories/').$model->image-
>baseName.'.'.$model->image->extension;
            $model->image->saveAs($path);
            $model->attachImage($path);
        }
        return $this->redirect(['view', 'id' => $model->idCategory]);
    }
}

```



```

return $this->render('update', [
    'model' => $model,
]);
}

/**
 * Deletes an existing CategoryOfFood model.
 * If deletion is successful, the browser will be redirected to the 'index' page.
 * @param integer $id
 * @return mixed
 * @throws NotFoundHttpException if the model cannot be found
 */
public function actionDelete($id)
{
    $this->findModel($id)->delete();

    return $this->redirect(['index']);
}

/**
 * Finds the CategoryOfFood model based on its primary key value.
 * If the model is not found, a 404 HTTP exception will be thrown.
 * @param integer $id
 * @return CategoryOfFood the loaded model
 * @throws NotFoundHttpException if the model cannot be found
 */
protected function findModel($id)
{
    if (($model = CategoryOfFood::findOne($id)) !== null) {
        return $model;
    }

    throw new NotFoundHttpException(Yii::t('app', 'The requested page does not exist.));
}

public function actionCategories()

```

```

{
    $array = CategoryOfFood::getAll();
    return $this->render('categories', ['CategoryOfFood'=>$array]);
}

public function actionProducts($idCategory)
{
    $query = FoodInformation::find()->where(['CategoryOfFood_idCategory'=> $idCategory])-
>orderBy([
        'foodName'=>SORT_ASC
    ]);

    $count = $query->count();

    $FoodInformation = $query->all();

    $data['FoodInformation'] = $FoodInformation;

    return $this->render('products',['FoodInformation' => $data['FoodInformation']]);

}

public function actionProduct($idFood)
{
    $FoodInformation = FoodInformation::findOne($idFood);

    return $this->render('product', compact('FoodInformation'));
}

}

```

#### Листинг В.4 – Модель продукта питания

```

class FoodInformation extends \yii\db\ActiveRecord
{
    /**
     * {@inheritdoc}
     */

    public static function tableName()

```

```

{
    return 'foodinformation';
}

/**
 * {@inheritdoc}
 */
public function rules()
{
    return [
        [['CategoryOfFood_idCategory'], 'required'],
        [['CategoryOfFood_idCategory'], 'integer'],
        [['foodName'], 'string', 'max' => 70],
        [['CategoryOfFood_idCategory'], 'exist', 'skipOnError' => true, 'targetClass' =>
Categoryoffood::className(), 'targetAttribute' => ['CategoryOfFood_idCategory' => 'idCategory']],
    ];
}

/**
 * {@inheritdoc}
 */
public function attributeLabels()
{
    return [
        'idFood' => 'Id Food',
        'foodName' => 'Наименование',
        'CategoryOfFood_idCategory' => 'Category Of Food Id Category',
    ];
}

/**
 * @return \yii\db\ActiveQuery
 */
public function getCategoryOfFoodIdCategory()
{
    return $this->hasOne(Categoryoffood::className(), ['idCategory' =>
'CategoryOfFood_idCategory']);
}

```

```

/**
 * @return \yii\db\ActiveQuery
 */
public function getFoodinformationHasBases()
{
    return $this->hasMany(FoodinformationHasBase::className(), ['FoodInformation_idFood'
=> 'idFood']);
}

/**
 * @return \yii\db\ActiveQuery
 */
public function getFoodinformationHasVitamins()
{
    return $this->hasMany(FoodInformationHasVitamins::className(),
['FoodInformation_idFood' => 'idFood']);
}

/**
 * @return \yii\db\ActiveQuery
 */
public function getVitaminsIdVitamins()
{
    return $this->hasMany(Vitamins::className(), ['idVitamin' => 'Vitamins_idVitamin'])-
>viaTable('foodinformation_has_vitamins', ['FoodInformation_idFood' => 'idFood']);
}

/**
 * @return \yii\db\ActiveQuery
 */
public function getIntakeinformations()
{
    return $this->hasMany(IntakeInformation::className(), ['FoodInformation_idFoodInfo' =>
'idFood']);
}

/**

```

```

* @return \yii\db\ActiveQuery
*/
public function getMacromicroelementsHasFoodinformations()
{
    return $this->hasMany(MacroMicroElementsHasFoodInformation::className(),
['FoodInformation_idFood' => 'idFood']);
}

/**
* @return \yii\db\ActiveQuery
*/
public function getMacroMicroElementsIdElements()
{
    return $this->hasMany(MacroMicroElements::className(), ['idElements' =>
'MacroMicroElements_idElements']->viaTable('macromicroelements_has_foodinformation',
['FoodInformation_idFood' => 'idFood']);
}
public function getCategoryOfFood()
{
    return $this->hasOne(CategoryOfFood::className(),['idCategory' =>
'CategoryOfFood_idCategory']);
}

public function saveCategoryOfFood($category_id)
{
    $categoryOfFood = CategoryOfFood::findOne($category_id);
    if($categoryOfFood != null)
    {
        $this -> link('categoryOfFood', $categoryOfFood);
        return true;
    }
}

public static function getAll()
{
    $data = self::find()->all();
    return $data;
}

```

```
}
```

## Листинг В.5 – Представление для продукта питания

```
<?php
use yii\helpers\Html;
use yii\widgets\LinkPager;
use yii\db\ActiveQuery;
use yii\db\ActiveRecord;
$this->title = Yii::t('app', 'Product');
$this->params['breadcrumbs'][] = $this->title;

?>

<h1><?= $FoodInformation->foodName ?></h1>
<p> Содержание нутриентов на 100 гр. продукта: </p>
<table class="tftable" border="1">
    <tr><th>Нутриент</th><th>Количество</th></tr>
<?php foreach ($FoodInformation->foodinformationHasBases as $bas):
    ?>
    <tr><td>
        <?php
        echo $bas->baseIdBase->name;
        ?>
    </td>
    <td>
        <?php
        echo $bas->count;

endforeach; ?>
    </td></tr>
</table>
<table class="tftable" border="1">
    <tr><th>Витамин</th><th>Количество</th></tr>
    <?php foreach ($FoodInformation->foodinformationHasVitamins as $vit):
        ?>
```

```

<tr><td>
    <?php
    echo $vit->vitaminsIdVitamins->Name;
    ?> </td>
<td>
    <?php
    echo $vit->Count;

    endforeach; ?>
</td></tr>
</table>

```

```

<table class="tftable" border="1">
    <tr><th>Макро/Микроэлемент</th><th>Количество</th></tr>
    <?php foreach ($FoodInformation->macromicroelementsHasFoodinformations as $mm):
    ?>

    <tr><td>
        <?php
        echo $mm->macroMicroElementsIdElements->elementName;
        ?>
    </td>
    <td>
        <?php
        echo $mm->Count;

        endforeach; ?>
    </td></tr>
</table>

```