

Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего профессионального образования

«Уральский федеральный университет имени первого Президента России  
Б. Н. Ельцина»

Физико-технологический институт

Кафедра технической физики

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой ТФ

\_\_\_\_\_ В. И. Токманцев

« \_\_\_\_\_ » июня 2017 г.

**ПРОГРАММНЫЙ ИНТЕРФЕЙС ДЛЯ  
СИСТЕМЫ АНАЛИЗА ТЕПЛОВЫХ УТЕЧЕК**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

090302 000000 04 ПЗ

Руководитель проф., к. т. н.

Рогович В. И.

Нормоконтролёр доц., к. т. н.

Ковалёв В. В.

Студент гр. Фт-430206

Молодых А. А.

Екатеринбург

2017

# **Реферат**

Пояснительная записка: 55 страницы.

# Содержание

Нормативные ссылки . . . . .	4
Обозначения и сокращения . . . . .	5
Введение . . . . .	6
1 Проблематика . . . . .	8
1.1 Основные термины и понятия . . . . .	8
1.2 Технология поиска информации . . . . .	8
1.3 Литературно-аналитический обзор . . . . .	8
1.4 Аналоги системы и их оценка . . . . .	11
1.4.1 MyHEAT . . . . .	11
1.4.2 National Heat Map . . . . .	12
1.4.3 HEAT . . . . .	13
1.4.4 blomURBEX . . . . .	14
1.4.5 Jersey Heat Loss Map . . . . .	14
1.4.6 Оценка аналогов . . . . .	15
1.5 Критика прототипа . . . . .	21
1.6 Уточнение целей и задач . . . . .	21
1.7 Результаты и выводы по главе 1 . . . . .	22
2 Моделирование . . . . .	23
2.1 Концептуальная модель . . . . .	23
2.2 Системно-структурная модель . . . . .	25
2.3 Функционально-структурная модель . . . . .	28
2.4 Алгоритмическая модель . . . . .	29
2.4.1 Алгоритмическая модель прототипа . . . . .	30
2.4.2 Алгоритмическая модель предлагаемого решения . . . . .	32
2.5 Результаты и выводы по главе 2 . . . . .	34

3	Проектирование предлагаемого решения . . . . .	36
3.1	Внешнее проектирование . . . . .	36
3.2	Внутреннее проектирование . . . . .	36
3.3	Результаты и выводы по главе 3 . . . . .	43
4	Инженерная реализация . . . . .	44
4.1	Описание реализации . . . . .	44
4.1.1	Примеры запросов и ответов API . . . . .	44
4.1.2	Экранные формы . . . . .	48
4.2	Результаты и выводы . . . . .	51
	Список использованных источников . . . . .	51
A	Функционально-структурные модели . . . . .	56
	Приложение А Функционально-структурные модели . . . . .	56
B	Техническое задание . . . . .	60
	Приложение Б Техническое задание . . . . .	60

## **Нормативные ссылки**

- В пояснительной записке использованы следующие стандарты:
- ГОСТ 19.701-90 ЕСПД. Схемы алгоритмов, программ, данных и систем;
- ГОСТ 34.602-89. Техническое задание на создание автоматизированной системы;
- РД 50-34.698-90. Автоматизированные системы. Требования к содержанию документов;
- ГОСТ Р ИСО/МЭК 12207-99 ИТ. Процессы жизненного цикла программных средств;
- ГОСТ Р 50.1.028–2001. Методология функционального моделирования.

## Обозначения и сокращения

В пояснительной записке использованы следующие обозначения и сокращения:

API	— Application Programming Interface;
CSE	— Center for Sustainable Energy;
CSV	— Comma-Separated Values;
EES	— Energy Efficient Solutions;
GEOBIA	— Geographic Object Based Image Analysis;
HTTP	— HyperText Transfer Protocol;
MVC	— Model-View-Controller;
NHM	— National Heat Map;
OGC	— Open Geospatial Consortium;
REST	— Respesentational State Transfer;
TURN	— Thermal Urban Road Normalization;
UML	— Unified Modeling Language;
VGI	— Volunteered geographic information;
БПЛА	— беспилотный летальный аппарат;
ГИС	— геоинформационная система;
ЖКХ	— жилищно-коммунальное хозяйство;
ИК съёмка	— инфракрасная съёмка;
МЭА	— международное энергетическое агентство;
ООП	— объектно-ориентированное программирование;
ПО	— программное обеспечение;
CATУ	— система анализа тепловых утечек.

## Введение

Согласно аналитическим данным, в России расход тепловой энергии на отопление многоквартирных домов составляет примерно 45% от объёма всех энергетических ресурсов страны [1]. Такое повышенное по сравнению с мировым значением можно объяснить тем, что большая часть территории страны расположена в северных областях, характеризующихся холодным климатом. По оценкам, проводимым организацией МЭА, в мировом масштабе процент потребления тепловой энергии для отопления зданий составляет 32-33% [2]. Прогноз этого показателя в долгосрочной перспективе показывает его рост в ближайшие несколько десятков лет при большинстве сценариев увеличения суммарной площади помещений [5].

В связи с этим актуальна задача повышения энергоэффективности зданий. Существует ряд причин возникновения тепловых потерь в помещениях, в частности имеют место тепловые утечки, происходящие в различных областях помещений (стены, крыши, окна, элементы вентиляции и пр.) [3, 4].

Для решения этой задачи применяют различные методы анализа теплопотерь. Наиболее распространенным на сегодняшний день считается использование тепловизионного сканирования зданий, которое позволяет не только дать точную оценку уровня тепловых утечек, но и обнаружить источники самих теплопотерь. Существует большое число частных компаний, занимающихся тепловизионным анализом помещений и выполняющих эту работу вручную. В этой связи возникла тенденция к созданию автоматизированных систем контроля тепловых утечек в городских зданиях. Потенциал таких систем, несомненно, высок, поскольку они способны накапливать в себе большие объёмы данных, которые можно подвергнуть статистическому анализу и на этой основе получать полезную информацию для частных владельцев домов, коммунальных организаций и городских служб.

Целью данной работы является проведение анализа имеющихся в мире систем контроля тепловых утечек, выбор прототипного решения среди этих систем и разработка улучшенной системы на его основе путём исправления выявленных недостатков.

# 1 Проблематика

## 1.1 Основные термины и понятия

В пояснительной записке применяют следующие термины с соответствующими определениями:

**Теплоэнергетическая эффективность здания** — способность здания поддерживать необходимую температуру воздуха в помещении за счёт его конструкции и изоляционных свойств внешнего материала в целях наиболее эффективного расходования тепловой энергии.

**Программный интерфейс** (англ. Application Programming Interface, API) — компонент сервиса или приложения, предоставляющий доступ к их функциям для программ сторонних разработчиков (клиентских приложений).

**Клиентское приложение** — программа, реализующая доступ конечных пользователей системы к её ресурсам путём обмена запросами и ответами с программным интерфейсом сервиса.

**Добровольная географическая информация** (англ. Volunteered Geographic Information, VGI) — общее название средств создания, сбора и распространения географических данных, добровольно предоставленных пользователями. VGI - частный случай т.н. *пользовательского контента*, используется в некоторых ГИС.

**Тепловизор** — устройство для наблюдения за распределением температуры исследуемой поверхности.

## 1.2 Технология поиска информации

## 1.3 Литературно-аналитический обзор

На сегодняшний день активно разрабатываются и поддерживаются веб-сервисы, использующие для своей работы данные геоинформационных систем.

Одна из разновидностей таких сервисов — **системы анализа тепловых утечек (САТУ)**, предоставляющие информацию об уровне энергетической эффективности зданий.

САТУ позволяют пользователям узнать оценки теплоэнергетической эффективности выбранных зданий; в некоторых случаях также возможно увидеть локализацию утечки тепла. Оценка эффективности основывается на результатах анализа тепловизионных изображений, сделанных наземно или с использованием воздушной съемки.

Большинство САТУ работают со снимками крыш зданий. Это можно объяснить следующими причинами:

- простота и доступность процесса воздушной съемки, возможность автоматизации за счёт использования БПЛА;
- масштабируемость системы за счет универсальности алгоритмов анализа зданий;
- большой охват обследуемой территории за относительно короткое время съемки;
- достаточный уровень достоверности результатов для невысоких зданий, для которых рассеивание тепла незначительно.

Такой подход позволяет получить оценку уровня теплоэффективности для каждого здания в исследуемом районе без значительного вмешательства человека в этот процесс. При обследовании изображений автоматически учитывается различная вспомогательная информация с целью повышения точности оценки. Так, в [10.3390/rs3071380] системе HEAT, использующей ИК-изображения, данные со спутников об излучении тепла и технологию TURN (нормализация по температуре дорожной поверхности), конечная оценка включает в себя не только результат анализа распределения температуры, но и внешние условия, такие как температура почвы, элемент нагрева поверхности крыш

в зависимости от материала и пр. Автоматизированность, кроме того, обеспечивается алгоритмами обнаружения жилых объектов на фотографиях, последующей сегментацией изображения и идентификацией этих объектов.

Тем не менее, более качественный и информативный анализ требует тщательного подхода к съёмке анализируемого объекта. Съёмка фасадов зданий может дать больше полезной информации, чем съёмка крыш с воздуха. Так, при частном исследовании зданий бригады экспертов используют специальное ИК-оборудование для захвата всех деталей фасадов, поскольку в этом случае важно выявление причин утечек, что не всегда возможно сделать при помощи только снимков крыш. Существующие способы [10.1109/JURSE.2011.5764737, 10.21611/qirt.2014.041, 10.1109/URS.2009.5137681], позволяющие с определённой долей вероятности выделить, а в некоторых случаях даже сформулировать [10.1016/j.eswa.2008.02.043] причины утечек, работают именно со снимками фасадов зданий. В целом, для оценки тепловой энергоэффективности невысоких зданий достаточно изображений поверхности кровли, однако при работе с более высокими зданиями, в том числе с многоэтажными, таких снимков не будет достаточно для объективного результата.

В этом случае возникает вопрос о том, как автоматизировать процесс съёмки фасадов зданий. Беспилотная съёмка позволяет быстро получать данные при съёмке крыш, но в применении к фасадам зданий в городской местности такой метод реализовать практически невозможно по ряду причин (технические и юридические причины). Проявляющаяся в настоящее время в геоинформационных системах тенденция к использованию VGI - добровольно предоставляемой географической информации - способна решить эту проблему [10.1007/s10708-007-9111-y]. Под термином VGI подразумеваются принципы сбора, обработки и распространения геоданных, предоставляемых лично владельцами (пользователями, имеющими к ним доступ) [wikipedia]. Такая практика

используется в проектах OpenStreetMap, Google Earth, WikiMapia, Panoramio и др. С развитием доступности Интернет по всему миру этот подход получения данных становится неоспоримо эффективным. В этой связи он может обеспечить САТУ более актуальными сведениями, а также значительно увеличить количество анализируемых объектов.

Для сбора и отправки VGI-данных можно использовать любое устройство, обладающее необходимыми датчиками и имеющее выход в Интернет. Такой подход используется, например, в системах регистрации сейсмической активности, работающих с данными акселерометров пользовательских устройств. Для этой цели хорошо подходят смартфоны, так как они мобильны и их использование широко распространено.

**Смартфон** — сложное многофункциональное устройство, как правило, снабженное встроенной фото- и видеокамерой, модулем Wi-Fi, датчиками положения в пространстве, ускорения. Существуют портативные камеры инфракрасной съемки, представляющие из себя модули, подключаемые к смартфону. Используя эти устройства, можно сделать распределенную систему сбора информации, добровольно предоставляемой пользователями смартфонов.

## **1.4 Аналоги системы и их оценка**

### **1.4.1 MyHEAT**

**MyHEAT** - система, разработанная исследовательским университетом Калгари, Канада. Система представлена веб-интерфейсом, позволяющим просмотреть тепловую карту местности, полученную в результате анализа тепловизионных аэроснимков зданий, исходные изображения зданий, а также узнать оценки их теплоэнергетической эффективности по шкале от 1 до 10 баллов, где 1 балл соответствует низкой эффективности, а 10 - высокой[\*]. Для визуального представления используется картографический сервис Google Maps с наложением слоев с информацией об утечках тепла. При построении карты используются

проприетарные алгоритмы, исключающие микроклиматическую изменчивость и геопространственные эффекты, а также учитывающие материал крыш и количество растительности на них, что повышает точность построения карт [\*]. Интерфейс системы представлен на рисунке 1.1.



Рисунок 1.1 – Внешний вид интерфейса системы MyHEAT

#### 1.4.2 National Heat Map

**National Heat Map** - разработка Центра по устойчивой энергетике (CSE) в Великобритании. Этот проект существует с 2010 года, данные в нём постоянно обновляются [\*]. НХМ позволяет получать данные по адресам зданий, в открытом доступе предоставляет интерактивную тепловую карту с выбором различных слоёв по категориям зданий (производственные, коммерческие и жилые). Эти слои представляют собой цветные изображения областей по уровням тепловыделения.

Данные о распределении тепла в конкретных зданиях предоставляются административным органам, а также организациям, имеющим специальную лицензию. Отличительная особенность данной системы - проработанный веб-интерфейс, который позволяет рассматривать карту по разным слоям одновременно, выделять интересующие области. Имеется возможность вывода резуль-

татов статистических расчётов по выбранной области в различных форматах (табличные, такие как CSV, диаграммы и графики). [\*] Такое решение подходит для случаев, когда нужен охват территории в масштабе целого континента. Однако оно не позволяет увидеть оценки для конкретных зданий в открытом виде, предоставляя лишь усреднённые показатели для комплексов зданий или отдельных улиц, районов и т. п.

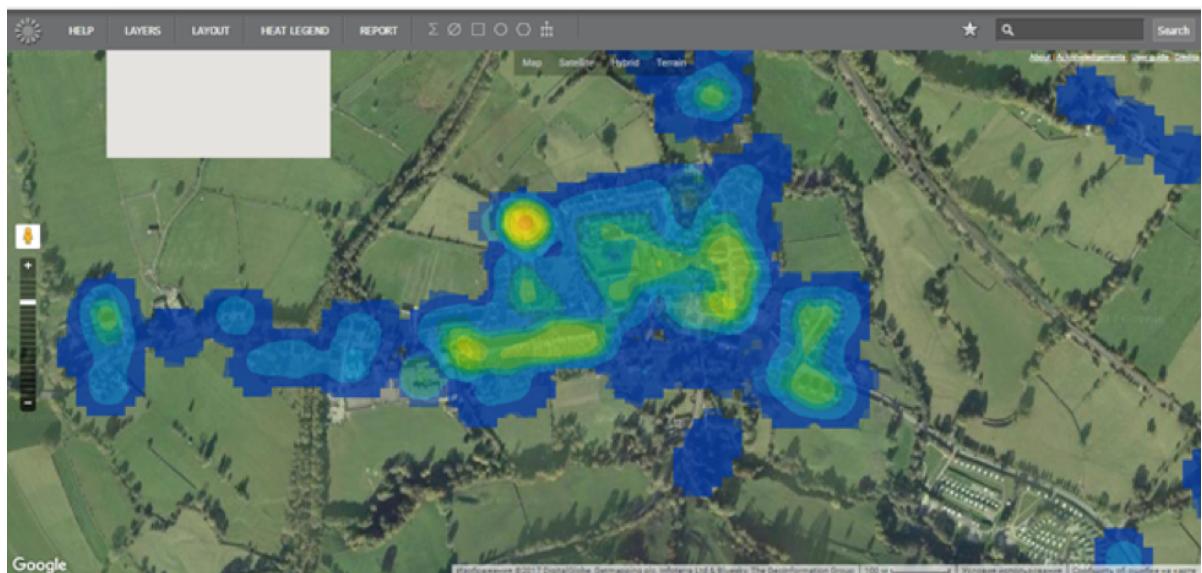


Рисунок 1.2 – Внешний вид интерфейса системы National Heat Map

#### 1.4.3 НЕАТ

**НЕАТ** - веб-ориентированная система анализа потерь тепла, использующая тепловые аэроснимки. Используются стандарты OGC (Open Geospatial Consortium) для геопространственных сервисов.[\*], картографический сервис Google Maps с возможностью выбора слоя тепловых утечек. Слой представляет из себя совокупность зданий, каждое из которых окрашено в соответствии с присвоенным ему одним из десяти классов энергетической эффективности, основанных на средней температуре крыши всех зданий. Система предоставляет такую информацию, как:

- минимальная и средняя температуры поверхности крыши выбранного здания;

- три наиболее горячие точки на поверхности крыши;
- суточную стоимость отопления дома и количество выделяющегося при этом углекислого газа в зависимости от вида топлива (газ/уголь/возобновляемые источники энергии);
- количество топлива, его стоимость и величина выделяемого углекислого газа, которые можно сократить при условии уменьшения температуры крыши до минимальной.

Важной особенностью сервиса является использование решений Geographic Object Based Image Analysis (GEOBIA), позволяющих в автоматическом или полуавтоматическом режиме генерировать готовые к внедрению в ГИС полигоны домов, анализируя непосредственно тепловые аэроснимки, а не используя имеющиеся городские кадастровые данные. Это играет существенную роль, когда кадастровые данные недоступны, и позволяет получить доступ к информации через веб-страницу в течение недели после сбора данных.

#### **1.4.4 blomURBEX**

**blomURBEX** - разработка норвежской компании Blom. Представляет из себя веб-сервис, предоставляющий доступ к карте, на которую добавлены слой информации об утечках тепла и оценка, присваиваемая теплоэнергетической эффективности здания. Можно производить поиск по адресу, также предусмотрена навигация по карте. Есть возможность нанесения на карту дополнительных графической объектов и текста, сохранения результатов в файл.

#### **1.4.5 Jersey Heat Loss Map**

Интерактивная карта, разработанная службой энергоэффективности (EES) острова Джерси, ориентирована не только на городские коммунальные службы, но и на самих владельцев домов, позволяя им увидеть оценку результатов тепловизионного анализа по установленной шкале уровней тепловых утечек.

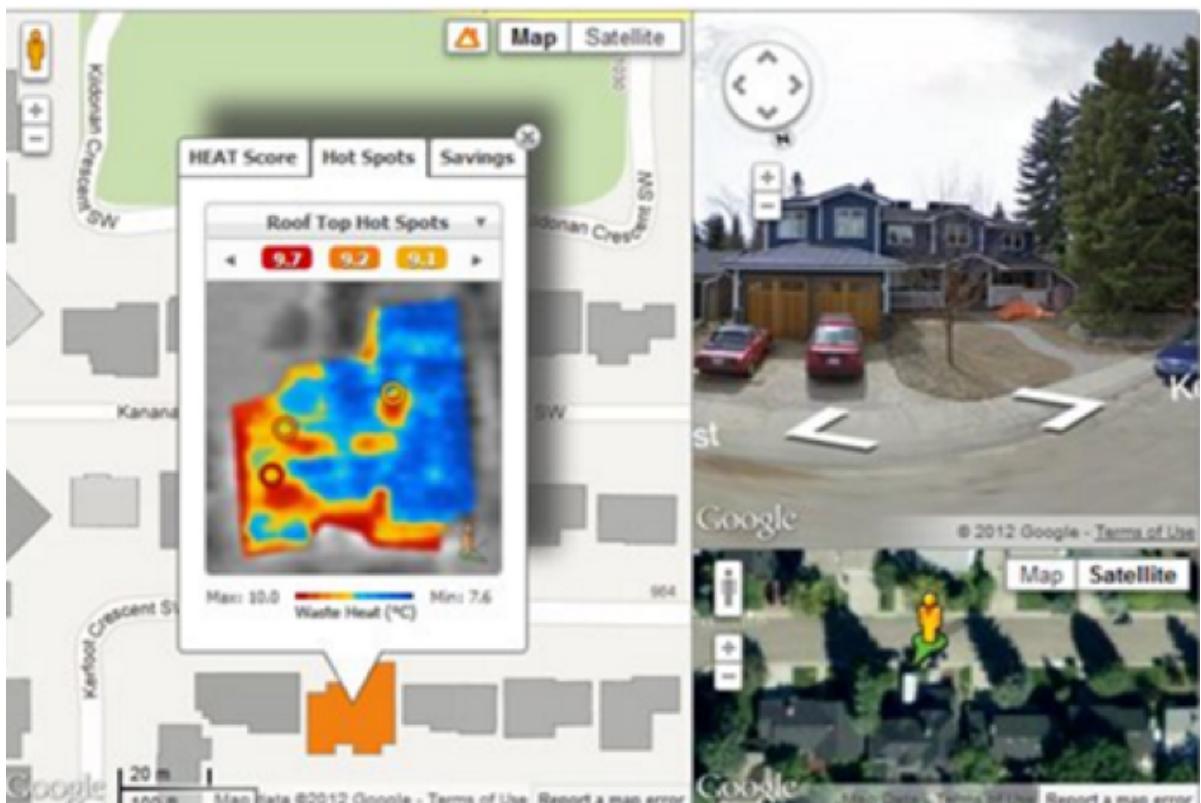


Рисунок 1.3 – Внешний вид интерфейса системы НЕАТ

Сами здания на карте представлены прямоугольниками разных цветов, соответствующих уровням этой шкалы. По каждому объекту можно получить такую информацию, как тип здания, индекс тепловых потерь и его текстовое описание. Из недостатков данной системой существенными являются отсутствие возможности увидеть изображения распределения тепла и получить статистику в произвольном масштабе.

#### 1.4.6 Оценка аналогов

Для сравнения аналогов были выбраны критерии, по которым будет производиться оценка. Кортежная модель оценки аналогов имеет вид:

$$OA = \langle O(Y\Phi), O(PT), O(P\vartheta), O(GP), O(KC), O(CP), O(PZ); R \rangle,$$

где ОА — оценка аналога,  $O(Y\Phi)$  — оценка учета факторов,  $O(PT)$  — оценка предоставления информации о распределении температур,  $O(P\vartheta)$  — оценка

расчёта энергии,  $O(\Gamma\Pi)$  — оценка генерации полигонов,  $O(KC)$  — оценка количества слоёв,  $O(SP)$  — оценка выдачи статистических расчётов,  $O(PZ)$  — оценка поиска зданий,  $R$  - матрица связи.

**Учёт факторов.** Первый критерий – учёт факторов – подразумевает “выравнивание” результатов с учётом изменения микроклимата от одного здания к другому, учёт материалов крыш, использование алгоритмов повышения достоверности информации. Данный критерий оценивается следующим образом:

$$I_{UF} = \begin{cases} 1, & \text{если учитывается микроклимат и материал крыши;} \\ 0.5, & \text{если используются статистические данные;} \\ 0, & \text{если дополнительные факторы не учитываются.} \end{cases}$$

**Распределение температур.** Критерий оценивается в зависимости от того, предоставляет ли выбранный аналог информацию о распределении температур по поверхности крыш зданий или нет:

$$I_{PT} = \begin{cases} 1, & \text{если система предоставляет информацию о распределении;} \\ & \text{температуры по обследуемой поверхности здания;} \\ 0, & \text{если система не предоставляет такую информацию.} \end{cases}$$

**Расчет энергии.** Критерий оценивается в зависимости от предоставления системой информации о количестве энергии, затрачиваемой на отопление

дома:

$$I_{\text{РЭ}} = \begin{cases} 1, & \text{если система предоставляет информацию о количестве энергии,} \\ & \text{затрачиваемой на отопление дома;} \\ 0, & \text{если система не предоставляет такую информацию.} \end{cases}$$

**Генерация полигонов.** Оценка производится на основе возможности системы создавать полигоны зданий, используя распознавание объектов и не прибегая к кадастровой информации.

$$I_{\text{ГП}} = \begin{cases} 1, & \text{если система генерирует полигоны;} \\ 0, & \text{если система не генерирует полигоны.} \end{cases}$$

**Количество доступных слоев.** При отображении карты на нее могут накладываться слои с различной вспомогательной информацией, статистическими данными и выбранными фильтрами. Оценка этого критерия производится следующим образом:

$$I_{\text{КС}} = \begin{cases} 1, & \text{если имеется три и более слоёв;} \\ 0.5, & \text{если имеется два слоя;} \\ 0, & \text{если имеется один слой.} \end{cases}$$

**Статистические расчеты.** Оценка этого критерия производится в зависимости от возможности предоставления системой статистических расчетов:

$$I_{\text{СР}} = \begin{cases} 1, & \text{если система предоставляет статистические расчёты по выбранной;} \\ & \text{территории с учётом параметров, характеризующих здания и специфи-} \\ 0, & \text{если система не предоставляет такую информацию.} \end{cases}$$

**Поиск зданий.** Оценка критерия производится следующим образом:

$$I_{CP} = \begin{cases} 1, & \text{если предусмотрена возможность поиска зданий по адресу;} \\ 0, & \text{если такая возможность не предусмотрена.} \end{cases}$$

Оценка аналогов будет производиться по следующей формуле:

$$\begin{aligned} OA = & O(Y\Phi) \cdot \alpha(Y\Phi) + O(PT) \cdot \alpha(PT) + O(P\Theta) \cdot \alpha(P\Theta) + \\ & + O(\Gamma\Pi) \cdot \alpha(\Gamma\Pi) + O(KC) \cdot \alpha(KC) + O(CP) \cdot \alpha(CP) + O(PZ) \cdot \alpha(PZ), \end{aligned}$$

где  $\sum_{i=1}^n \alpha_i = 1$ ,  $\alpha_i$  — весовой коэффициент  $i$ -го критерия.

### Определение весовых коэффициентов критериев.

При методе попарных сравнений используется шкала словесных определений уровня важности, где каждому определению ставится в соответствие число (таблица 1.1)

Таблица 1.1 – Шкала уровней важности

Уровень важности	Количественное значение
Равная важность	1
Промежуточное значение между равенством и слабым превосходством	2
Умеренное превосходство	3
Промежуточное значение между слабым и сильным превосходством	4
Существенное или сильное превосходство	5
Промежуточное значение между сильным и значительным превосходством	6
Значительное (большое) превосходство	7
Промежуточное значение между значительным и абсолютным превосходством	8
Абсолютное превосходство	9

Таблица 1.2 – Оценка весовых коэффициентов критериев

	УФ	РТ	РЭ	ГП	КС	СР	ПЗ	$\Sigma$	$\alpha$
УФ	1	2	3	4	5	6	8	29.00	0.26
РТ	1/2	1	2	4	6	7	9	29.50	0.26
РЭ	1/3	1/2	1	3	5	5	7	21.83	0.19
ГП	1/4	1/4	1/3	1	3	3	6	13.83	0.12
КС	1/5	1/6	1/5	1/3	1	2	5	8.90	0.08
СР	1/6	1/7	1/5	1/3	1/2	1	4	6.34	0.06
ПЗ	1/8	1/9	1/7	1/6	1/5	1/4	1	2.00	0.02
ЭР	1/9	1/9	1/8	1/7	1/6	1/6	1/2	1.32	0.01
$\Sigma\Sigma$								112.73	1

Оценка аналогов производилась с использованием методики Томаса Саати. В таблице 1.2 представлена матрица попарного сравнения критериев.

В таблице 1.3 приведены оценки аналогов по выбранным критериям.

Таблица 1.3 – Показатели критериев аналогов

	УФ	РТ	РЭ	ГП	КС	СР	ПЗ	ЭР
Jersey Heat Loss Map	0	0	0	0	0	0	1	0
HEAT	1	1	1	1	0,5	0	0	0
MyHEAT	1	1	1	1	0	0	1	0
BlomURBEX	0	0	0	0	0,5	0	1	1
National Heat Map	0.5	0	1	0	1	1	1	1

Взвешенные оценки аналогов представлены в таблице 1.4.

Таблица 1.4 – Оценки аналогов по критериям

	УФ	РТ	РЭ	ГП	КС	СР	ПЗ	ЭР	$\alpha$
	0.26	0.26	0.19	0.12	0.08	0.06	0.02	0.01	1.00
Jersey Heat Loss Map	0	0	0	0	0	0	0.02	0	0.02
HEAT	0.26	0.26	0.19	0.12	0,04	0	0	0	0.87
MyHEAT	0.26	0.26	0.19	0.12	0	0	0.02	0	0.85
BlomURBEX	0	0	0	0	0,04	0	0.02	0.01	0.07
National Heat Map	0.13	0	0.19	0	0.08	0.06	0.02	0.01	0.49

Наиболее высокую оценку получил аналог HEAT, целесообразно рассматривать его в качестве прототипа.

## **1.5 Критика прототипа**

В результате оценки аналогов наилучший результат получила система HEAT Университета Калгари, Канада. Основные ее преимущества - использование алгоритмов повышения достоверности результатов, автоматическая генерация полигонов зданий.

Существенный недостаток системы HEAT — отсутствие возможности работы с пользовательскими данными. Не предусмотрена функция загрузки снимков и вспомогательных данных с пользовательских устройств (в частности, со смартфонов) и их последующая обработка. Доработка этих возможностей позволит наполнить систему более полной и актуальной информацией в соответствии с принципами VGI.

## **1.6 Уточнение целей и задач**

Глобальная цель работы — увеличить количество источников данных для обследования жилых зданий с целью оценки их тепловой энергоэффективности.

Локальная цель — разработать программный интерфейс (API), позволяющий подключать к системе приложения сторонних разработчиков на различных устройствах.

Задачи, которые необходимо решить для достижения поставленной цели:

- выполнить аналитический обзор имеющихся в мире информационных систем, выполняющих задачу оценки тепловой энергоэффективности;
- провести анализ используемых в этих системах технологий, которые применяются для получения исходных данных;
- обнаружить среди обозреваемых систем и технологий наиболее совершенные с точки зрения поставленной в данной работе цели и выделить прототип разрабатываемой системы;
- разработать пакет моделей на создание улучшенной информационной системы;

- выполнить внешнее и внутреннее проектирование искомой системы (API САТУ);
- реализовать программное решение искомой системы (API САТУ).

## **1.7 Результаты и выводы по главе 1**

Проведён поиск информации об используемых решениях в области систем анализа тепловых утечек жилых зданий. Выполнен обзор литературных источников и существующих аналогов таких систем, сформулировано их общее описание. Среди аналогов на основе оценки наиболее существенных факторов выявлен прототип. На основании критики предложена доработка прототипа. Определены цели и задачи данной работы.

## **2 Моделирование**

### **2.1 Концептуальная модель**

**Система анализа тепловых утечек** — программный комплекс, выполняющий следующие основные функции:

- оценка энергетической эффективности зданий городской застройки;
- выдача результатов оценивания по запросу;
- обработка ИК снимков;
- упрощение процесса проведения ИК съёмки пользователями.

#### **Пути реализации основных функций:**

- обнаружение критичных областей и вычисления средних значений показателей распределения тепла по ИК снимкам;
- предоставление веб-доступа к результатам обследования;
- геометрическая коррекция изображений и учета внешних условий съёмки;
- программное обеспечение процесса проведения ИК съемки.

#### **Структурная основа реализации:**

- методы статистического анализа;
- способы визуализации данных;
- алгоритмы нормализации ИК снимков по инвариантным признакам спутниковых снимков;
- клиент-серверная архитектура.

**Направленность функционирования системы:** обеспечение информационной поддержки процесса обнаружения, обследования, контроля и устранения тепловых утечек.

**Цель функционирования системы:** повышение качества обследования жилых объектов на предмет тепловой энергоэффективности.

**Программный интерфейс (API)** для системы анализа тепловых утечек городской застройки.

**а) Основные функции:**

- 1) сбор данных;
- 2) унификация данных, поступающих в систему анализа;
- 3) обеспечение их корректности;
- 4) обеспечение доступности данных для системы анализа;
- 5) предоставление результатов анализа клиентским приложениям.

**б) Пути реализации основных функций:**

- 1) приём пакетов данных от различных источников;
- 2) преобразование данных в одинаковый формат;
- 3) проверка пакетов входных данных на соответствие требованиям;
- 4) взаимодействие с БД системы анализа;
- 5) обработка внешних запросов на результаты анализа утечек.

**в) Структурная основа реализации:**

- 1) для функций а.1, а.5: сетевые протоколы обмена информацией;
- 2) для функции а.2: требования системы анализа тепловых утечек;
- 3) для функции а.3: методы фильтрации нежелательного контента;
- 4) для функции а.4: централизованный подход к управлению данными в СУБД.

**г) Направленность функционирования системы:** расширение географической области, охватываемой системой анализа и увеличение числа пользователей системы анализа.

**д) Цели функционирования системы:** предоставление набора функций, реализуемых системой анализа тепловых утечек, сторонним программам вне зависимости от их платформы и аппаратного обеспечения.

## 2.2 Системно-структурная модель

На рисунке 2.1 изображена системно-структурная модель системы анализа тепловых утечек. Внедрение в неё новых структурных элементов – программного интерфейса и мобильного приложения – приводит к изменению состава таких подсистем, как веб-приложение и модуль работы с ИК изображениями. Модели этих подсистем представлены на рисунках [2.2, 2.3] соответственно.

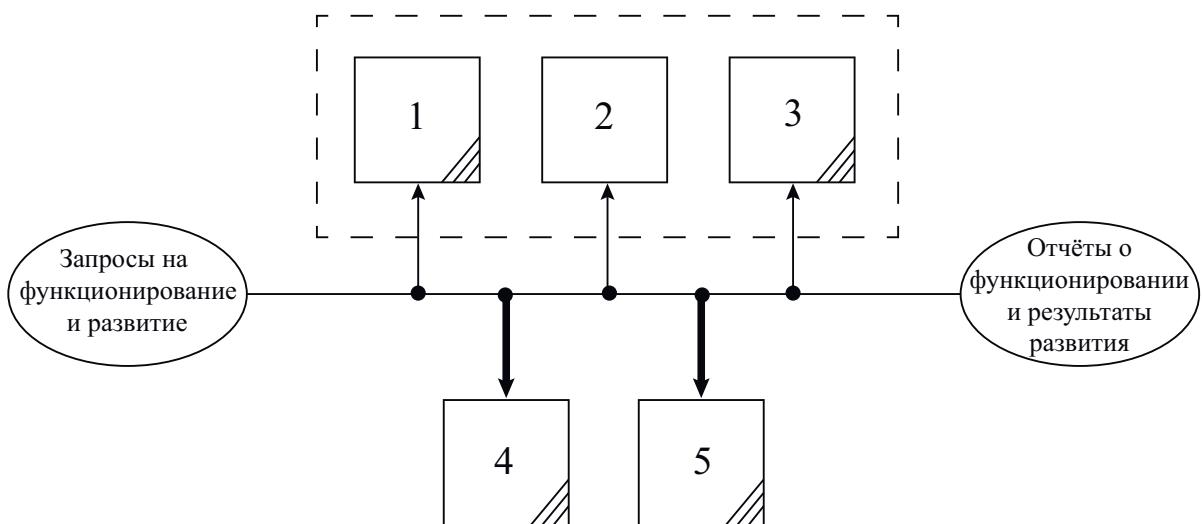


Рисунок 2.1 – Системно-структурная модель системы анализа тепловых утечек: 1 - веб-приложение, 2 - подсистема управления данными, 3 - модуль работы с ИК изображениями, 4 - мобильное приложение, 5 - программный интерфейс (API)

Структурные компоненты web-приложения представлены разделами, с которыми работают его пользователи (рисунок 2.2). Работа каждого раздела обеспечивается web-сервером и множеством программных сценариев, генерирующих динамические web-страницы, содержащие информацию, соответствующую названию раздела.

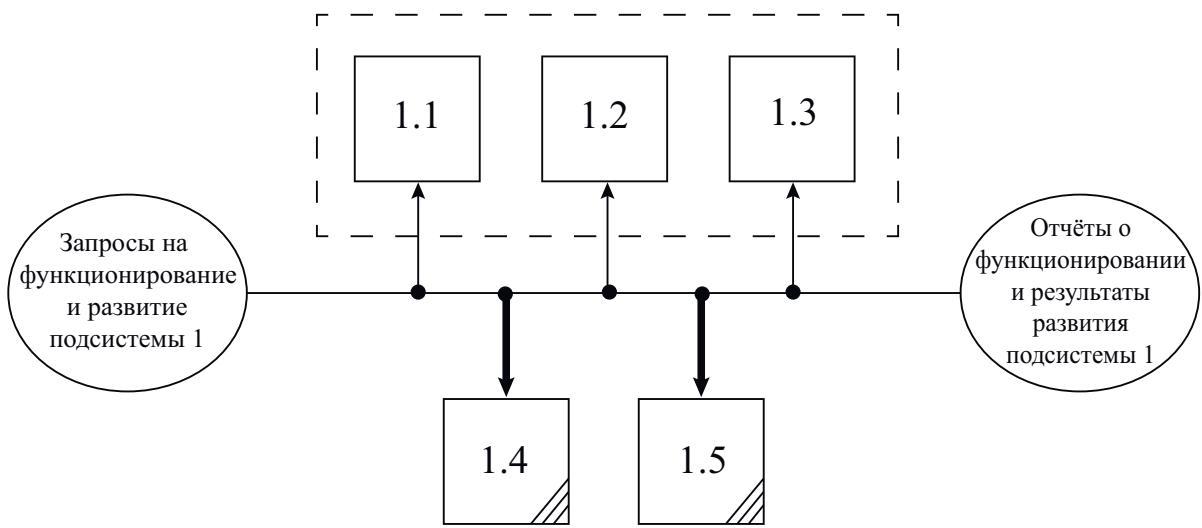


Рисунок 2.2 – Системно-структурная модель веб-приложения: 1.1 - блок отображения карты, 1.2 - блок отображения оценок энергоэффективности и энергозатрат, 1.3 - блок представления изображений, 1.4 - раздел «личного кабинета», 1.5 - раздел загрузки пользовательских данных

Компоненты модуля работы с ИК изображениями (рисунок 2.3) разделены по характеру выполняемых преобразований: подсистема 3.1 решает задачу распознавания зданий с ИК аэроснимков, описанную в [\*\*\*МуНЕАТ\*\*\*], процедуры обработки в подсистеме 3.2 устраняют отклонения, вызванные локальными изменениями климата, на снимках, подсистема 3.3 использует алгоритмы математической статистики для итоговых расчётов.

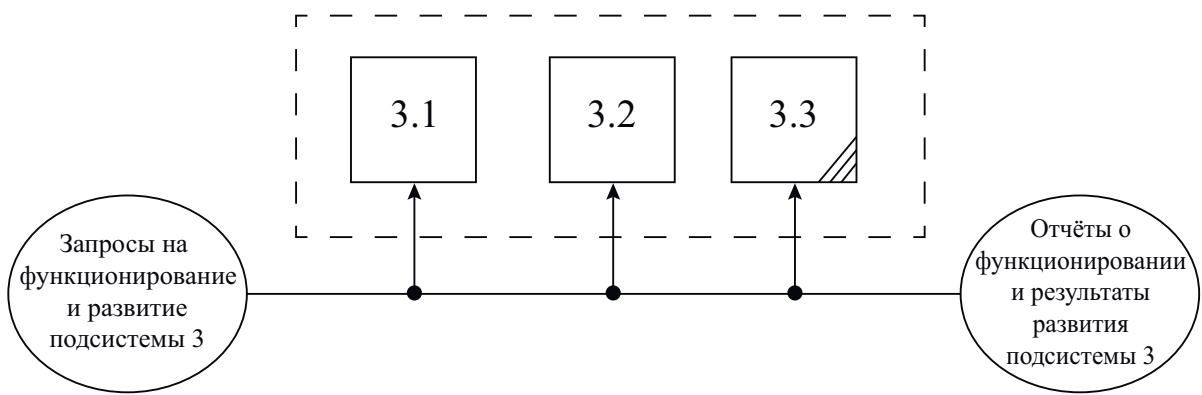


Рисунок 2.3 – Системно-структурная модель модуля работы с ИК изображениями: 3.1 - подсистема фотограмметрической обработки ИК снимков, 3.2 - подсистема коррекции по микроклиматическим условиям, 3.3 - подсистема расчёта оценки энергоэффективности

В связи с тем, что в систему анализа тепловых утечек внедряется новый вид съёмки, очевидно, что некоторые подсистемы претерпят изменения, которые отражены в алгоритмических моделях. Подсистемы 3.1 и 3.2 являются исключениями, поскольку для наземной съёмки отдельных зданий они не актуальны. Многие задачи обработки наземных снимков берут на себя программные клиенты - мобильные приложения.

В модели программного интерфейса системы анализа тепловых утечек, представленной на рисунке 2.4, в качестве подсистем прототипа были взяты стандартные компоненты, участвующие в работе большинства API относительно крупных программных комплексов.

В рамках системы анализа тепловых утечек специфика API заключается в наличии подсистемы 5.5. Это связано с характерными особенностями данных, поступающих в систему. Например, в систему могут поступать данные с ИК камер различных производителей, и, кроме того, данные различных типов съёмки. Расширение спектра возможных источников данных - одна из причин внедрения API в основную систему.

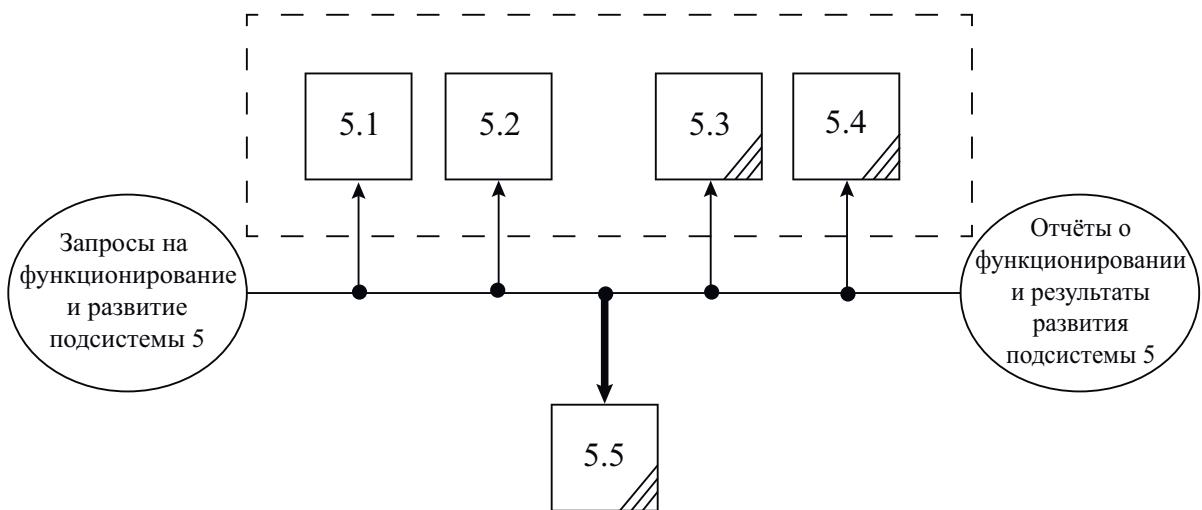


Рисунок 2.4 – Системно-структурная модель программного интерфейса: 5.1 - подсистема приёма запросов и отправки данных, 5.2 - подсистема аутентификации пользователей, 5.3 - подсистема формирования запросов к БД, 5.4 - подсистема валидации данных, 5.5 - подсистема унификации и форматирования данных

## 2.3 Функционально-структурная модель

Функционально-структурная модель предлагаемого решения построена при помощи пакета Computer Associates ERwin Process Modeler 7.3. Модели выполнены в соответствии с методологией IDEF0. Диаграммы модели приведены в приложении А.

Все функции, выполняемые системами анализа тепловых утечек, можно объединить в один функциональный блок “Оценить энергетическую эффективность зданий”, как показано на диаграмме, изображенной на рисунке А.1. Рассмотрение функциональной структуры системы в целом важно при моделировании разрабатываемого программного интерфейса, поскольку это даёт представление о взаимосвязи информационных потоков между смежными функциями всей системы, а также с её внешней средой.

Диаграмма декомпозиции 1 уровня включает в себя набор основных функций, которые выполняются подсистемами, перечисленными в системно-структурной модели А.2. Так, функциональный блок А3 выполняется программ-

ным интерфейсом (подсистема 5, рисунок 2.4). Некоторые подсистемы могут участвовать в работе нескольких функций. Например, за работу блоков А2 и А6 отвечает мобильное приложение (подсистема 3, рисунок 2.3).

Описание математических алгоритмов содержится в некоторых алгоритмических моделях, приведённых в разделе 2.4. Под клиентскими приложениями понимается ПО, которое отображает карты энергоэффективности зданий (в частности, мобильное и web-приложения).

На рисунке А.3 представлена диаграмма декомпозиции блока “*Отправить снимки на сервер*”. Все функциональные блоки в этой диаграмме задействуют API, поэтому она представляет наибольший интерес среди остальных блоков узла А0.

## **2.4 Алгоритмическая модель**

В разделе 2.4.1 приведены алгоритмы процессов работы системы до внедрения в неё новых подсистем. В разделе 2.4.2 представлены алгоритмы, в которых происходят наиболее существенные изменения после внедрения внедряемых подсистем, а также алгоритм работы разрабатываемого программного интерфейса.

## 2.4.1 Алгоритмическая модель прототипа

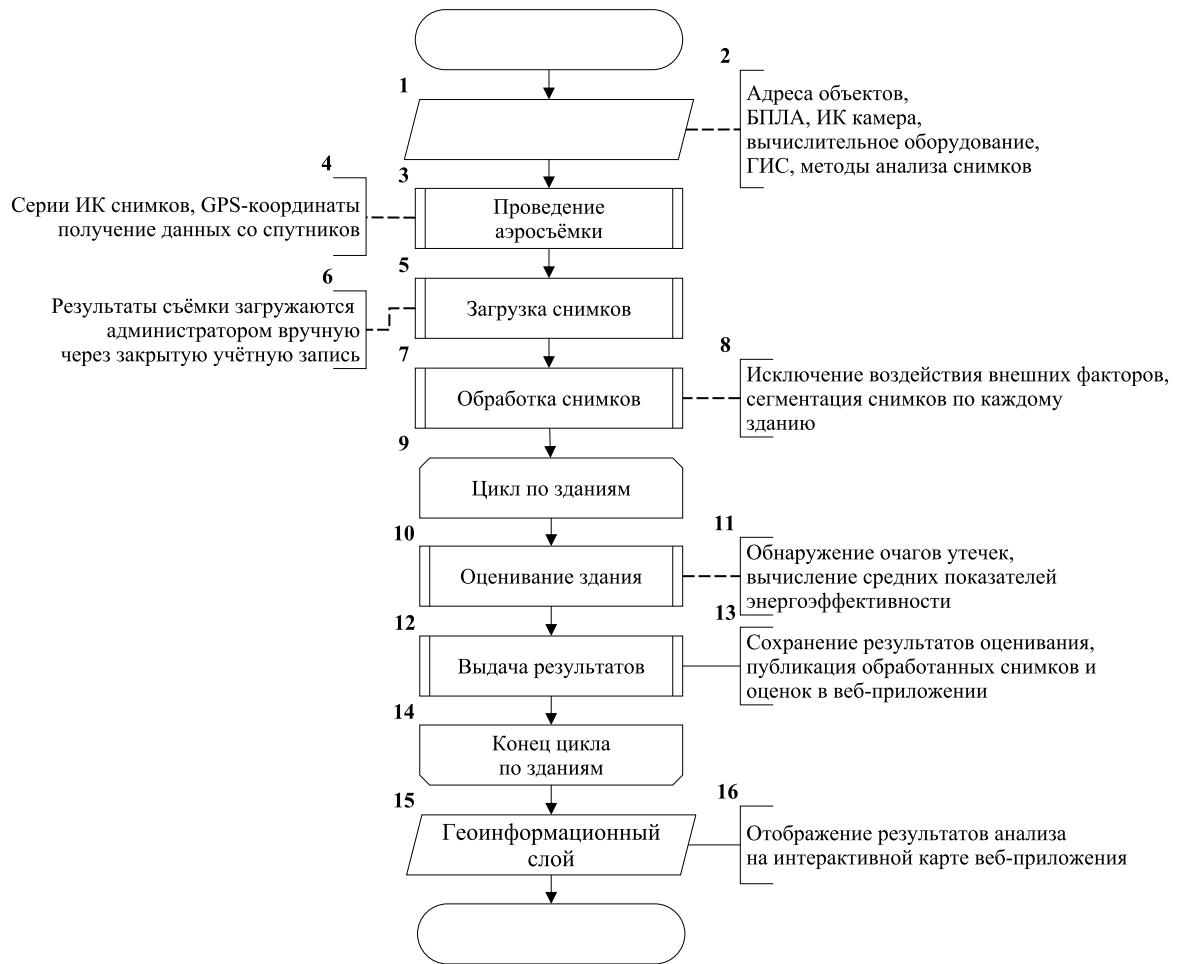


Рисунок 2.5 – Алгоритм исследования утечек в жилых зданиях (для прототипа)

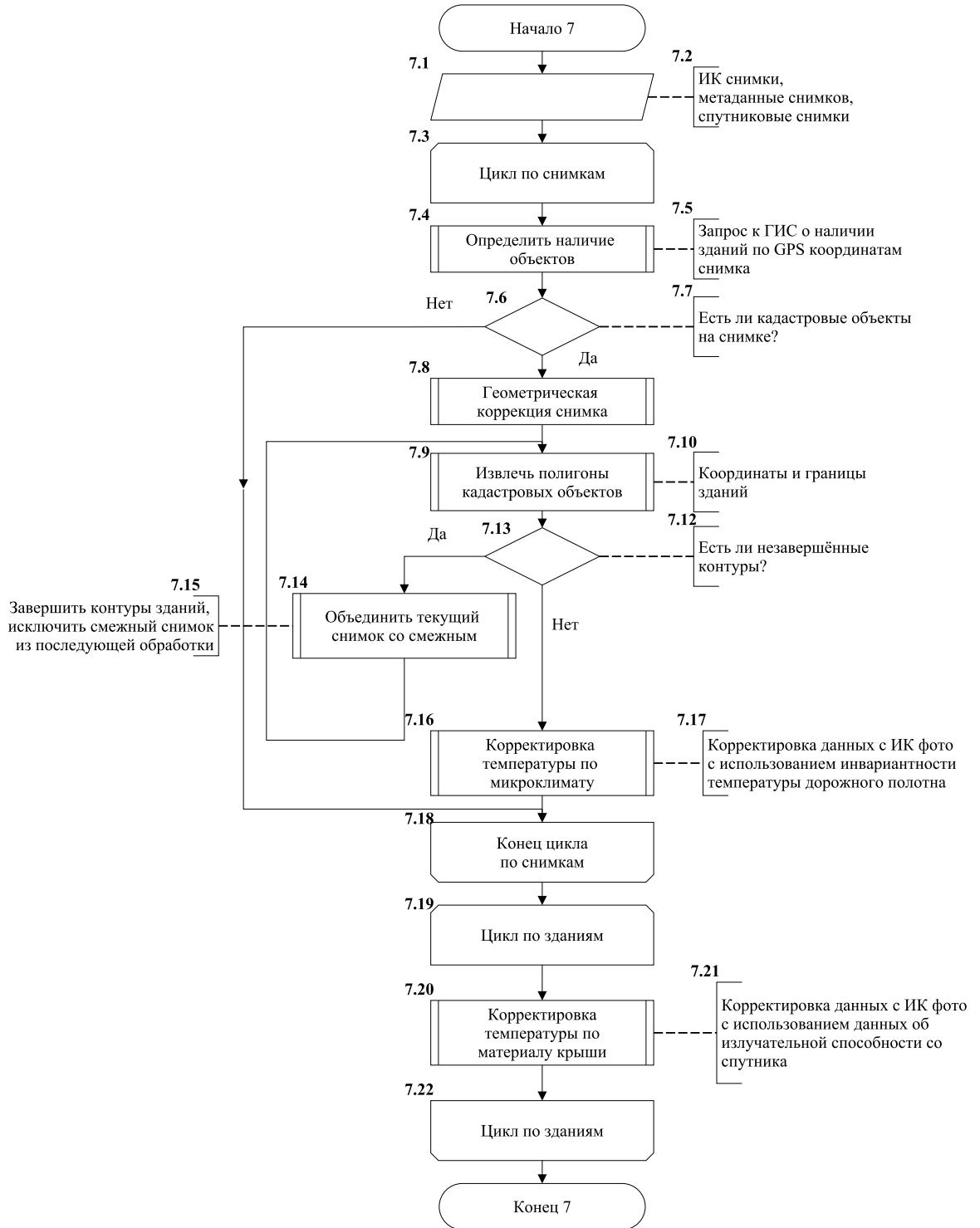


Рисунок 2.6 – Алгоритм обработки аэроснимков (для прототипа)

## 2.4.2 Алгоритмическая модель предлагаемого решения

В связи с тем, что предлагаемая система предполагает два вида съёмки, изменяется основной алгоритм исследования утечек жилых зданий для расчёта оценки их энергоэффективности (рисунок 2.9). Возникает проблема, связанная с тем, что одно здание может быть отснято несколько раз, причём разным способом. Для решения этой проблемы следует проводить интегральную (итоговую) оценку, представленную в алгоритме процессом 20. Более детальное описание этого процесса описывает алгоритм, приведённый на рисунке 2.8.

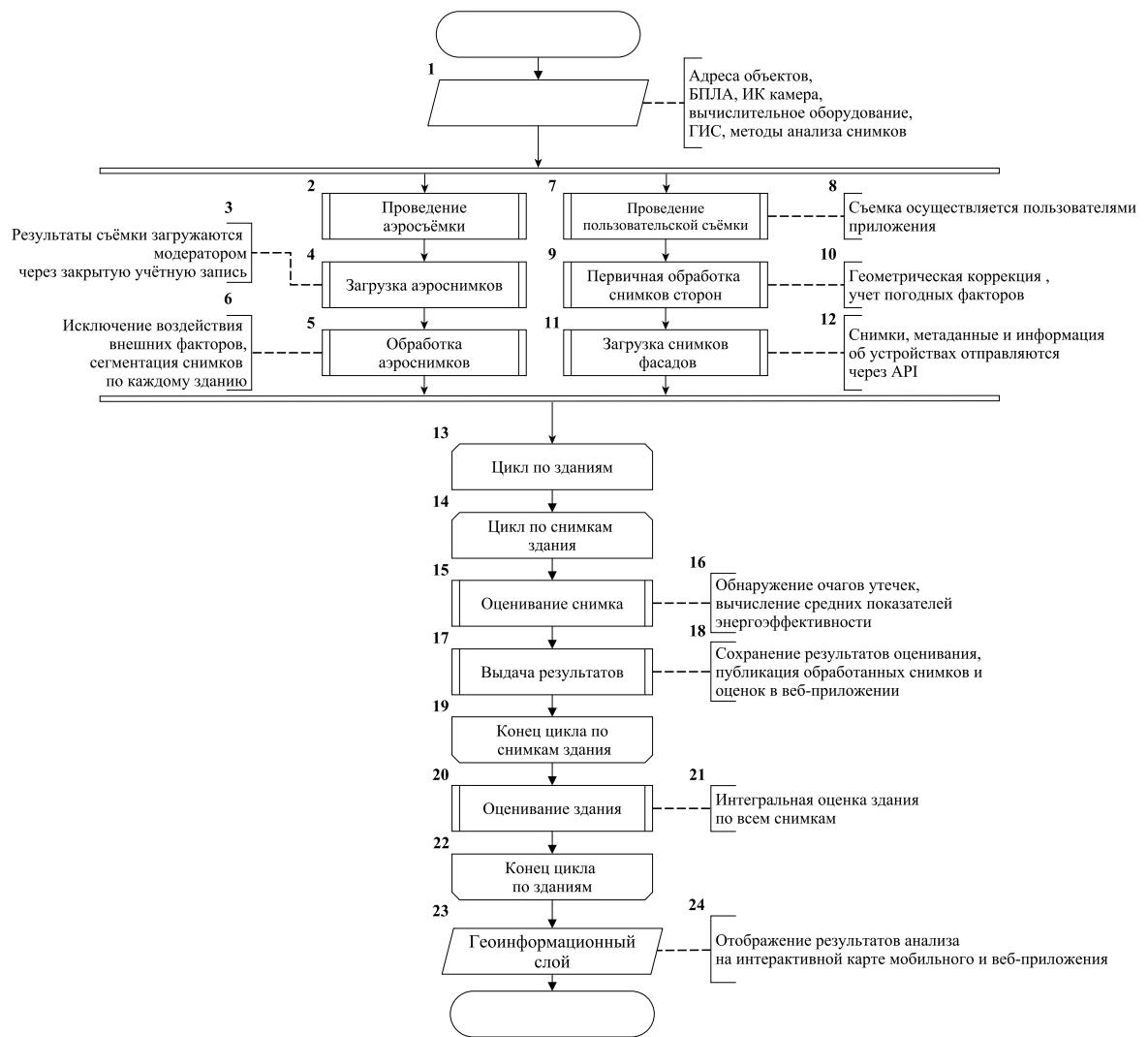


Рисунок 2.7 – Алгоритм исследования утечек в жилых зданиях (для предлагаемого решения)

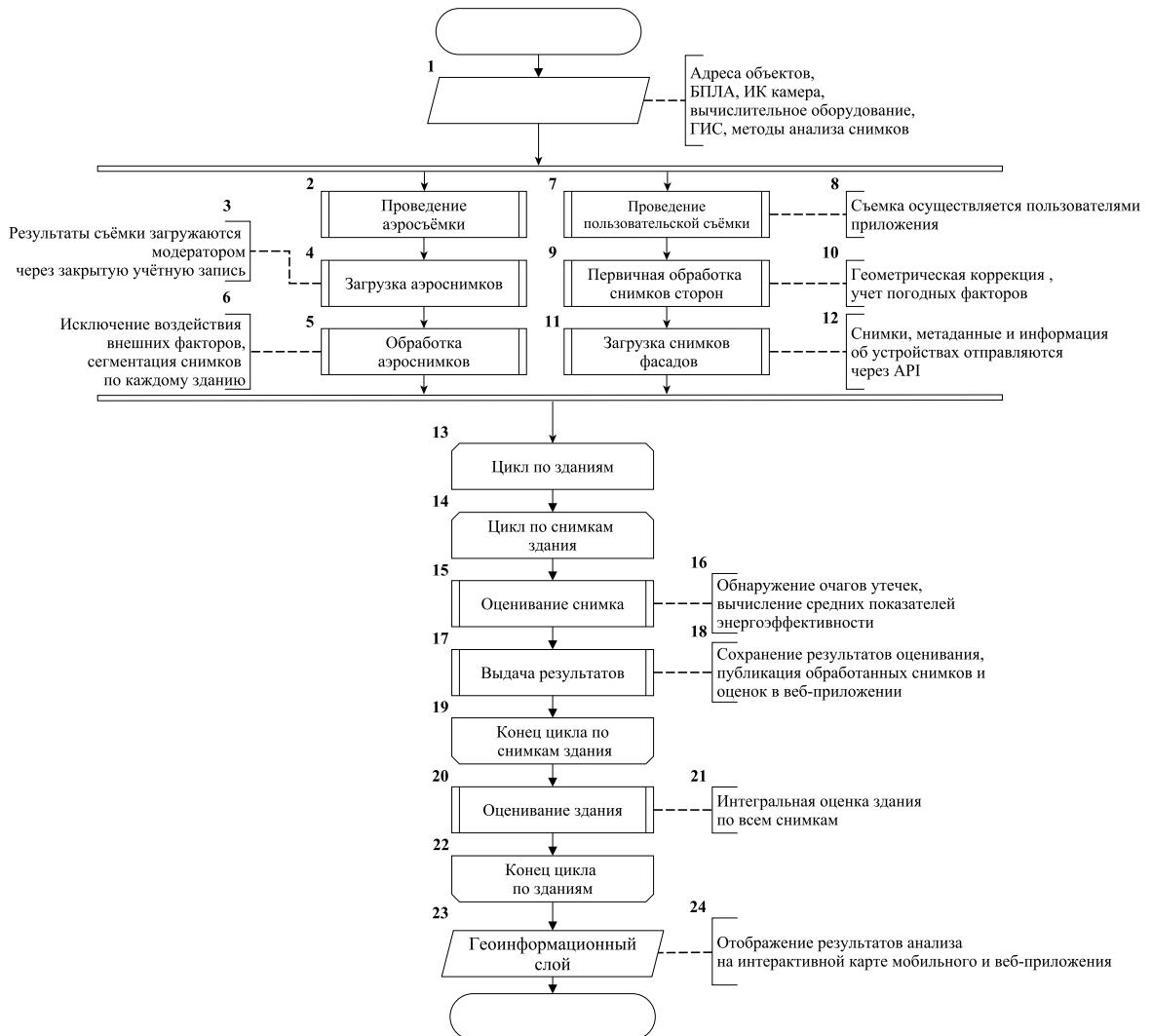


Рисунок 2.8 – Алгоритм итогового оценивания энергоэффективности (для предлагаемого решения)

После внедрения новых подсистем процедура получения пользователем информации из системы также претерпевает изменения. Сравнение алгоритмов работы с системой до изменений (рисунок 2.5) и после них (рисунок 2.9) позволяет их выделить.

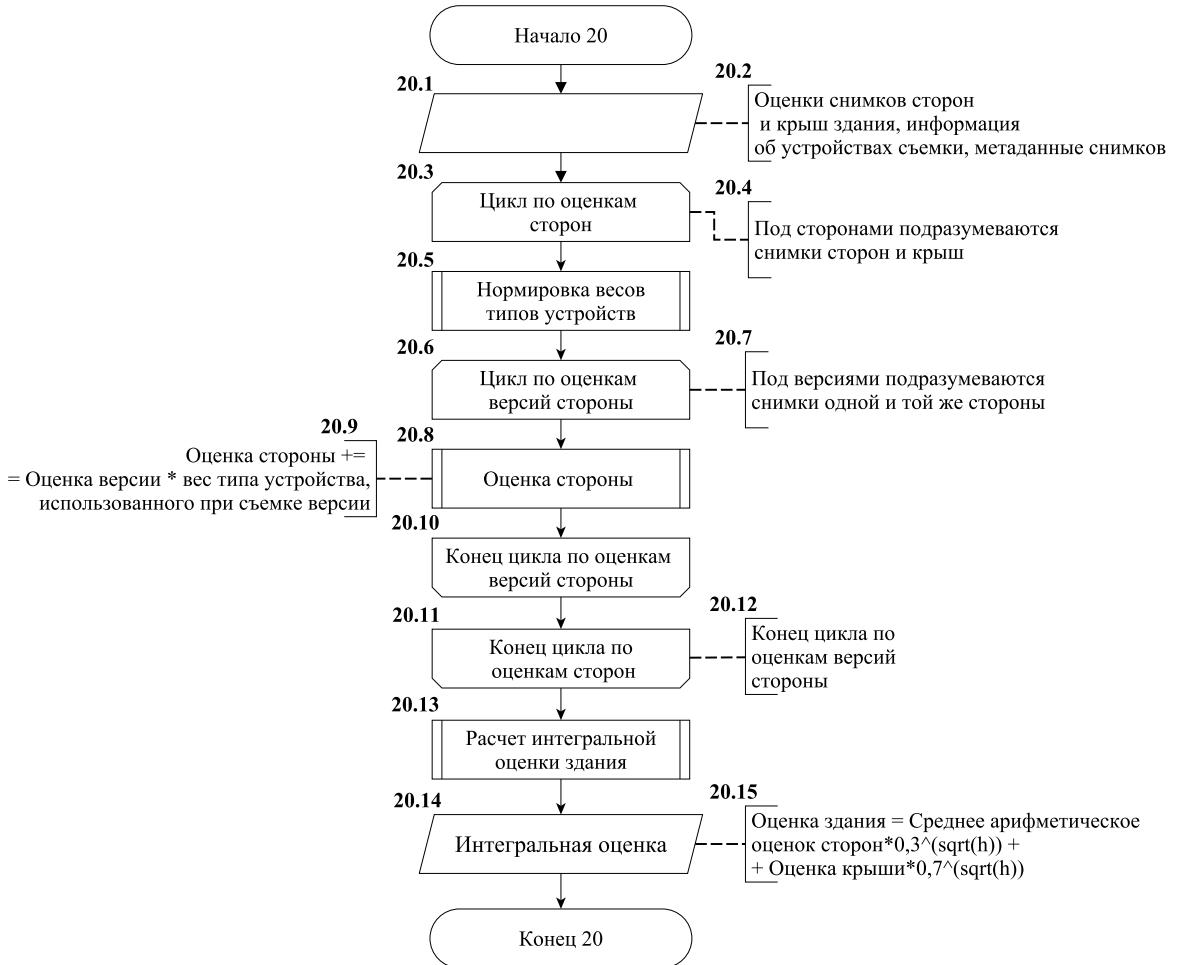


Рисунок 2.9 – Алгоритм использования системы в целях получения информации об энергоэффективности жилых зданий (для предлагаемого решения)

## 2.5 Результаты и выводы по главе 2

На основе информации, полученных в результате изучения материалов, приведённых в главе 1, были построены модели прототипа и предлагаемого решения. В целях выделения специфики между существующей системой и предлагаемым решением представлена структурно-функциональная модель и алгоритмы работы с подсистемами прототипа и новой системы.

Благодаря тому, что моделирование проводилось как для системы в целом, так и для разрабатываемого программного интерфейса, можно получить представление о том, как изменяются отдельные компоненты системы, как вза-

имодействуют информационные потоки смежных с API подсистем и внешней среды всей системы анализа тепловых утечек. Взаимосвязь между системой в целом и её подсистемами показана в концептуальной, системно-структурной и структурно-функциональной моделях.

Алгоритмы, представленные в разделе 2.4, имеют целью продемонстрировать ключевые изменения в работе компонентов системы, а также в работе пользователей с системой. Кроме того, смоделирован алгоритм работы разрабатываемого API, который отражает специфику на фоне аналогов в области разработки программных интерфейсов.

Таким образом, становится возможным формулирование конкретного списка требований, которым должны удовлетворять все подсистемы в новом решении на этапе проектирования и задач, которые необходимо выполнить при реализации нового решения в рамках данной работы.

# **3 Проектирование предлагаемого решения**

## **3.1 Внешнее проектирование**

В результате внешнего проектирования было получено техническое задание по ГОСТ 34.602-89 на разработку программного интерфейса (API) искомой системы, приведенное в приложении \*.

## **3.2 Внутреннее проектирование**

На основе требований, выдвинутых в техническом задании и результатов моделирования, приведённых в главе 2 проведено внутреннее проектирование API САТУ.

Согласно требованиям, API САТУ взаимодействует с клиентскими приложениями через Интернет. В этой связи разумно использовать инструменты, ориентированные на работу в Web.

Для разработки программного интерфейса системы анализа тепловых утечек в городских зданиях был выбран сценарный язык программирования PHP, поддерживающий концепцию ООП, поддерживающий концепцию объектно-ориентированного программирования. Основная причина выбора этого языка - распространённость его использования в web-проектах, в том числе для web-API, в силу наличия большого количества встроенных библиотек, ориентированных на разработку web-проектов.

Проектирование API выполнено на основе шаблона MVC (Model-View-Controller), который подразумевает деление программного кода на 3 отдельных компонента: *модель* (отвечает за обработку данных), *представление* (отвечает за получение и отправление данных) и *контроллер* (содержит управляющую логику, использует модели и представления). MVC позволяет сделать независимым код, отвечающий за работу с данными от кода, отвечающего за приём и

отправку запросов [\*].

В качестве подхода к проектированию взаимодействия клиентских приложений и компонентов системы, работающих на сервере, была взят архитектурный стиль REST. Ограничения, определённые в REST, позволяют создавать масштабируемые и унифицированные программные интерфейсы [\*]. REST использует в качестве протокола прикладного уровня HTTP.

Стандарт, который использует API для форматирования сообщений - JSON. Формат JSON подходит для передачи сообщений со сложной структурой и уместен при обмене данными между клиентскими приложениями и сервером.

Согласно требованиям в техническом задании (приложение Б), а также алгоритму итогового оценивания энергоэффективности (раздел 2.4.2), API взаимодействует с данными САТУ с помощью СУБД, а именно обеспечивает загрузку и выгрузку снимков зданий. Таким образом, API необходим доступ к базе данных САТУ. Для инженерной реализации была спроектирована та часть базы данных САТУ, которая будет задействована в работе API.

Проектирование базы данных выполнялось на основе ER-модели (“сущность-связь”). Соответствующая ER-диаграмма представлена на рисунке 3.1 и включает в себя основные сущности:

**Пользователь (user)** - хранит идентификационные данные пользователей, их контакты, а также некоторые технические данные для системы аутентификации.

**Здание (building)** - хранит данные из ГИС, необходимые для САТУ (в том числе результаты оценивания). Необходим для прикрепления к конкретным зданиям серий снимков.

**Снимок (snapshot)** - хранит данные, полученные пользователями с различных устройств, в том числе ИК изображения (в двоичном представлении),

координаты съёмки, а также данные, собранные со вспомогательных датчиков в момент съёмки здания устройством.

**Серия снимков** (`snapshot_series`) - используется для объединения снимков зданий, выполненных конкретным пользователем с помощью конкретного ИК устройства за определённый интервал времени. Это необходимо для правильного хранения версий снимков здания за счёт логических ограничений в структуре базы данных.

**ИК устройство** (`ir_device`) - позволяет централизованно хранить всю техническую информацию об устройствах съёмки (тепловизорах). Таким образом, у клиентов нет необходимости отсылать эти данные после каждой серии съёмки.

**Приложение** (`application`) - хранит идентификационные данные клиентских приложений, сведения об ограничениях доступа к функциям API и дополнительные сведения о его конфигурации.

**Ключ доступа** (`token`) - предназначен для хранения ключей доступа, привязанных к приложению, а также информации об этих ключах: тип доступа(ограниченный/неограниченный), статус(активен/неактивен). Хранение токенов необходимо для того, чтобы исключить дублирование при генерации новых.

**Журналы** (`log_user`, `log_application`) - используются для фиксирования некоторых действий пользователей и приложений в целях отладки клиентских программ.

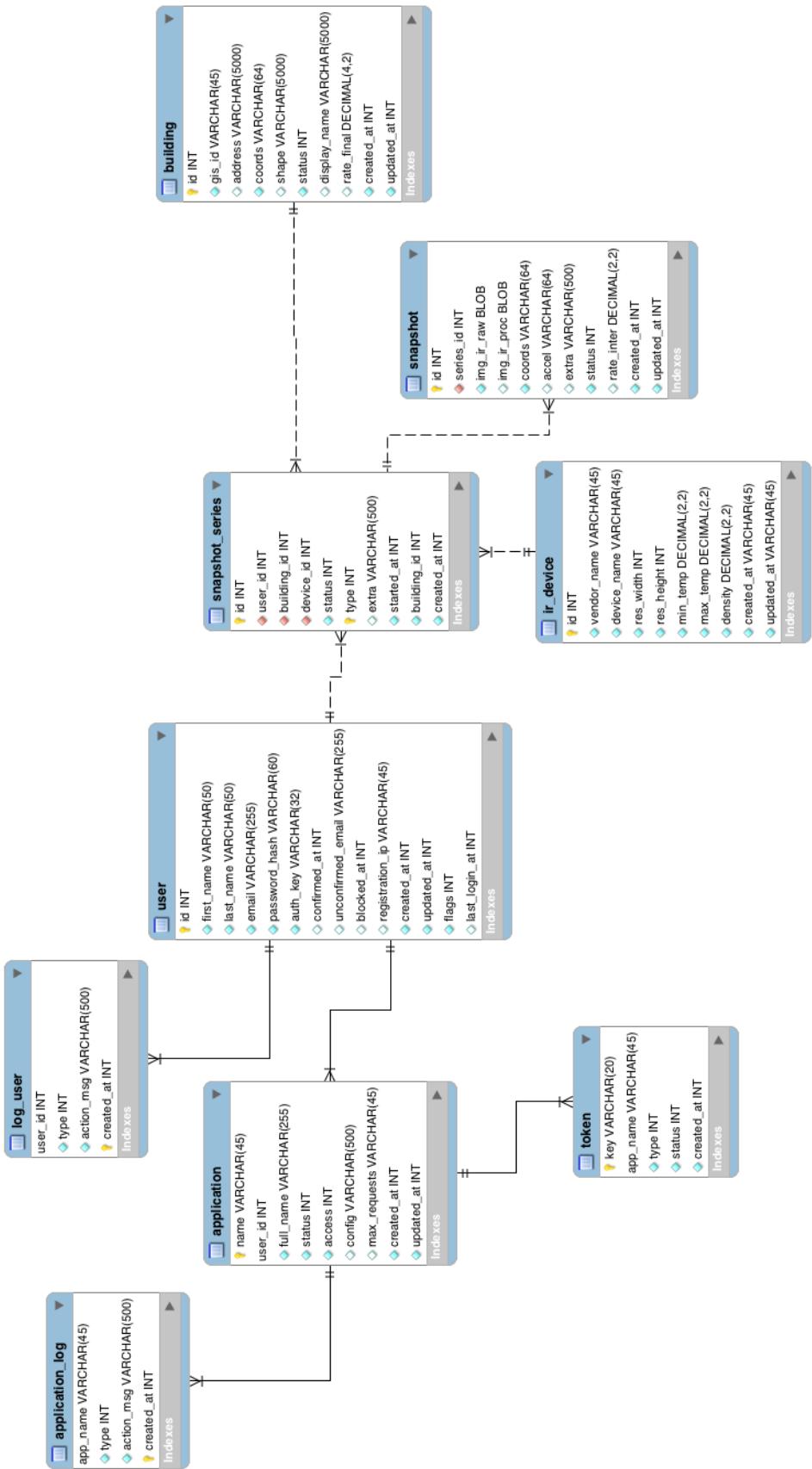


Рисунок 3.1 – Фрагмент схемы базы данных САТУ, используемой программным интерфрейсом

На основе полученной схемы базы данных была спроектирована архитектура взаимодействия клиентских приложений с API путём изображения ресурсов, предоставляемых API и ссылок на эти ресурсы. Схемы, получаемые в результате проектирования, полезны при написании программного кода, когда точно сформулированы запросы, их параметры, определены необходимые для работы функций классы. Кроме того, эти схемы можно использовать при создании руководства программиста, поскольку в основном описывает набор доступных запросов, входные и выходные данные, их формат и структуру.

При проектировании REST API используют диаграммы классов [ссылка]. Каждый ресурс характеризуется названием, идентификатором (URI), соответствующим HTTP-методом и набором параметров. Ресурсы на диаграмме изображены скругленными прямоугольниками. В качестве описателей входных и выходных данных ресурсов используют классы (выделены прямоугольной рамкой). Проектирование выполнено с помощью программного пакета Visual Paradigm 14.1, предусматривающего функцию генерации каркаса программного кода на основе построенных моделей.

На рисунке 3.2 представлена диаграмма классов, описывающая набор ресурсов, специфичных для САТУ. Классы и ресурсы, относящиеся к авторизации пользователей и приложений, в данной диаграмме опущены.

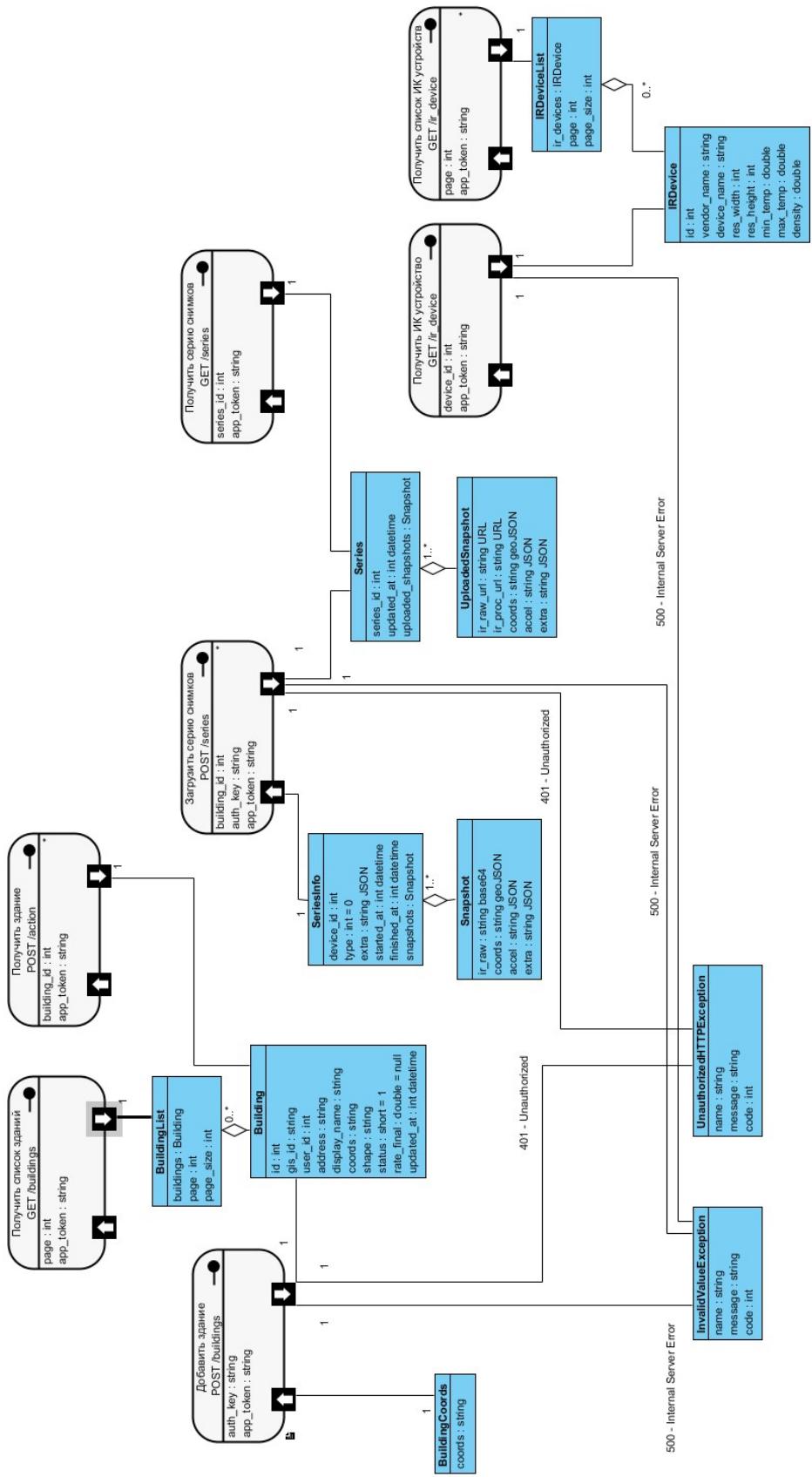


Рисунок 3.2 – Фрагмент диаграммы классов, описывающей структуру основной части API САТУ

Атрибуты некоторых классов в представленной диаграмме во многом совпадают с атрибутами сущностей в ER-диаграмме. Тем не менее, атрибуты классов показывают, что должен включать в себя тот или иной HTTP-запрос к ресурсу. При обращении к нему клиентская программа должна учитывать правильность соблюдения не только типов передаваемых данных, но и структуры отправляемого запроса.

Некоторые классы связаны между собой отношением агрегации. Например, чтобы загрузить снимки какого-либо здания в систему, требуется передать данные о съёмке (класс `SeriesInfo`): идентификатор ИК устройства (`device_id`), тип съёмки (`type`), а также несколько снимков (`snapshots`), где каждый снимок (класс `Snapshot`) описан четырьмя атрибутами: изображение, закодированное в формате base64, координаты снимка, данные встроенного акселерометра в формате JSON (`accel`) и т. п.

В данной диаграмме обозначены классы `InvalidValueException`, `UnauthorizedHTTPException`, которые используются для оповещения клиентов об ошибках при передаче невалидных данных и при некорректной аутентификации/авторизации. HTTP-ответы, содержащие сведения об ошибке, API помечает соответствующим статусом. В соответствии с REST [ссылка], коды ошибок необходимо использовать по назначению, когда это возможно.

### **3.3 Результаты и выводы по главе 3**

На основе полученных на этапе моделирования результатов сформулированы требования к разрабатываемому программному интерфейсу, определены программные средства и архитектурные принципы, которые легли в основу внутреннего проекта. Таким образом, были достигнуты следующие результаты:

- составлено Техническое задание на разработку системы (приложение Б);
- определён набор программных средств и инструментов для создания системы;
- разработаны решения по программной архитектуре системы.

## 4 Инженерная реализация

### 4.1 Описание реализации

Выбранные архитектурные принципы, стандарты и инструменты, утверждённые на этапе проектирования, поддерживаются программной платформой Yii 2 Framework. Данная платформа ориентирована на ускорение процесса разработки web-проектов, в том числе API по принципу REST, за счёт наличия большого числа программных модулей и автоматической генерации программного кода на основе схем баз данных. В результате внутреннего проектирования предлагаемого решения выяснилось, что эта платформа является наиболее подходящей.

#### 4.1.1 Примеры запросов и ответов API

Отправка запросов и получение ответов выполнялось с помощью среды тестирования и отладки API Postman. Каждое сообщение содержит заголовок и тело, содержимое которых определяется ресурсом в соответствии с диаграммой на рисунке 3.2.

Для отправки запроса на добавление здания необходимо передать API запрос, аналогичный представленному на рисунке 4.1. При этом возможны два сценария: здание успешно добавлено (рисунок 4.2), здание уже существует (рисунок 4.3).

```

1 POST /api/web/buildings?app_token=NMQXo1STUdCRTdC60GpV2RTNMQXo5S9ZbGUv40V0 HTTP/1.1
2 Host: hlcs:8080
3 Cache-Control: no-cache
4 Content-Type: multipart/form-data;
5 Authorization: Basic UjAxaVZpcEpFTzl0STdCRTNMQXo5SGp0V09ZV2V4bGU6
6
7 {
8     "lat": 56.822984,
9     "lon": 60.544541
10 }

```

Рисунок 4.1 – Пример запроса на добавление здания в САТУ

```

1 HTTP/1.1 200 OK
2 Connection: Keep-Alive
3 Content-Length: 1066
4 Content-Type: application/json; charset=UTF-8
5 Date: Sun, 11 Jun 2017 20:24:38 GMT
6 Keep-Alive: timeout=5, max=100
7 Location: http://hlcs:8080/api/web/buildings/16
8 Server: Apache
9 X-Powered-By: PHP/5.6.10
10
11 {
12     "id": 16,
13     "gis_id": "51687249",
14     "address": "\"house_number\": \"59 к3\", \"road\": \"улица Викулова\", \"neighbourhood\": \"Медгородок\", \"suburb\": \"ВИЗ\",
15     \"city_district\": \"Верх-Исетский район\", \"city\": \"Екатеринбург\", \"county\": \"городской округ Екатеринбург\",
16     \"state\": \"Свердловская область\", \"country\": \"РФ\", \"country_code\": \"ru\"}",
17     "display_name": "59 к3, улица Викулова, Медгородок, ВИЗ, Верх-Исетский район, Екатеринбург,
18     городской округ Екатеринбург, Свердловская область, Уральский федеральный округ, РФ",
19     "coords": "{\"lat\": \"56.8228379\", \"lon\": \"60.544569045849\"}",
20     "shape": "[[[60.5442835, 56.8229606], [60.5444476, 56.8228024], [60.54448, 56.8227711], [60.5446082, 56.8226475], [60.5446133, 56.8226425],
21     [60.5448528, 56.8227169], [60.5446905, 56.8228734], [60.544523, 56.823035], [60.5442835, 56.8229606]]]",
22     "status": 1,
23     "rate_final": null
24 }

```

Рисунок 4.2 – Успешный ответ к запросу на добавление здания в САТУ

```

1 HTTP/1.1 500 Internal Server Error
2 Connection: close
3 Content-Length: 1538
4 Content-Type: application/json; charset=UTF-8
5 Date: Sun, 11 Jun 2017 20:19:01 GMT
6 Server: Apache
7 X-Powered-By: PHP/5.6.10
8
9 {
10     "name": "Exception",
11     "message": "Здание уже существует в системе",
12     "code": 0,
13     "type": "yii\\base\\InvalidArgumentException"
14 }

```

Рисунок 4.3 – Ответ к повторному запросу на добавление того же здания

Некоторые ресурсы требуют указания ключа авторизации. Если таковые отсутствуют, клиентскому приложению будет возвращён ответ, представленный на рисунке 4.4.

```

1 HTTP/1.1 401 Unauthorized
2 Connection: Keep-Alive
3 Content-Length: 150
4 Content-Type: application/json; charset=UTF-8
5 Date: Sun, 11 Jun 2017 22:04:54 GMT
6 Keep-Alive: timeout=5, max=100
7 Server: Apache
8 WWW-Authenticate: Basic realm="api"
9 X-Powered-By: PHP/5.6.10
10
11 {
12     "name": "Unauthorized",
13     "message": "Your request was made with invalid credentials.",
14     "code": 0,
15     "status": 401,
16     "type": "yii\\web\\UnauthorizedHttpException"
17 }

```

Рисунок 4.4 – Ответ к запросу без ключа авторизации

```

1 GET /api/web/buildings?page=1&app_token=NMQXo1STUdCRTdC60GpV2RTNMQXo5S9ZbGU40V0 HTTP/1.1
2 Host: hlcs:8080
3 Content-Type: application/x-www-form-urlencoded
4 Cache-Control: no-cache

```

Рисунок 4.5 – Пример запроса на получение списка зданий

На рисунке 4.5 приведен пример запроса на получение одной страницы списка всех зданий, соответствующий ответ - на рисунке 4.6. Из заголовка ответа видно, что объём каждой страницы - 20 элементов. При необходимости получить другую страницу следует изменить параметр `page`.

Если требуется получить информацию только по одному конкретному зданию, то к URI предыдущего запроса следует добавить идентификатор здания (рисунок 4.7). В результате от сервера API придёт ответ, аналогичный примеру на рисунке 4.8.

```

1 HTTP/1.1 200 OK
2 Connection: Keep-Alive
3 Content-Type: application/json; charset=UTF-8
4 Date: Sun, 11 Jun 2017 20:10:20 GMT
5 Keep-Alive: timeout=5, max=100
6 Link: <http://hlcs:8080/api/web/buildings?page=1&app_token=NMQXo1STUdCRTdC60GpV2RTNMQXo5S9ZbGUv40V0>; rel=self
7 Server: Apache
8 Transfer-Encoding: chunked
9 X-Pagination-Current-Page: 1
10 X-Pagination-Page-Count: 1
11 X-Pagination-Per-Page: 20
12 X-Pagination-Total-Count: 15
13 X-Powered-By: PHP/5.6.10
14
15 [
16 {
17   "id": 1,
18   "gis_id": "51342964",
19   "address": "{\"house_number\": \"30\", \"pedestrian\": \"улица Вайнера\", \"suburb\": \"Юго-Западный\",",
20   | \"city_district\": \"Ленинский район\", \"city\": \"Екатеринбург\", \"county\": \"городской округ Екатеринбург\",
21   | \"state\": \"Свердловская область\", \"postcode\": \"620041\", \"country\": \"РФ\", \"country_code\": \"ru\"}",
22   "display_name": "30, улица Вайнера, Юго-Западный, Ленинский район, Екатеринбург, городской округ Екатеринбург",
23   | Свердловская область, Уральский федеральный округ, 620041, РФ",
24   "coords": "{\"lat\": \"56.8323468\", \"lon\": \"60.5958021064869\"}",
25   "shape": "[[60.595619, 56.8323957], [60.595674, 56.832237], [60.5959947, 56.8322703], [60.5959852, 56.8322979], [60.5959398, 56.832429],
26   | [60.595619, 56.8323957]]",
27   "status": 1,
28   "rate_final": "6.12"
29 },
30 {
31   "id": 2,
32   "gis_id": "52784364",
33   "address": "{\"house_number\": \"15\", \"pedestrian\": \"улица Вайнера\", \"suburb\": \"Юго-Западный\",",
34   | \"city_district\": \"Ленинский район\", \"city\": \"Екатеринбург\", \"county\": \"городской округ Екатеринбург\",
35   | \"state\": \"Свердловская область\", \"postcode\": \"620041\", \"country\": \"РФ\", \"country_code\": \"ru\"}",
36   "display_name": "15, улица Вайнера, Юго-Западный, Ленинский район, Екатеринбург, городской округ Екатеринбург",
37   | Свердловская область, Уральский федеральный округ, 620041, РФ",
38   "coords": "{\"lat\": \"56.83247265\", \"lon\": \"60.5968596382458\"}",
39   "shape": "[[60.5963924, 56.8325413], [60.5964744, 56.8322953], [60.5966891, 56.8323175], [60.5966767, 56.832348], [60.5969849, 56.832379],
40   | [60.5969453, 56.8324913], [60.5965895, 56.832454], [60.5965674, 56.8325192], [60.5967353, 56.8325362], [60.5967037, 56.8326293],
41   | [60.5966226, 56.832621], [60.5966084, 56.8326628], [60.5964329, 56.832645], [60.5964656, 56.8325487], [60.5963924, 56.8325413]]",
42   "status": 1,
43   "rate_final": "6.40"
44 }

```

Рисунок 4.6 – Ответ к запросу на получение списка зданий

```

1 GET /api/web/buildings/1?app_token=NMQXo1STUdCRTdC60GpV2RTNMQXo5S9ZbGUv40V0 HTTP/1.1
2 Host: hlcs:8080
3 Cache-Control: no-cache

```

Рисунок 4.7 – Пример запроса на получение информации о конкретном здании

```

1 HTTP/1.1 200 OK
2 Connection: Keep-Alive
3 Content-Length: 976
4 Content-Type: application/json; charset=UTF-8
5 Date: Sun, 11 Jun 2017 20:30:29 GMT
6 Keep-Alive: timeout=5, max=99
7 Server: Apache
8 X-Powered-By: PHP/5.6.10
9
10 {
11   "id": 1,
12   "gis_id": "51342964",
13   "address": "{\"house_number\": \"30\", \"pedestrian\": \"улица Вайнера\", \"suburb\": \"Юго-Западный\",",
14   | \"city_district\": \"Ленинский район\", \"city\": \"Екатеринбург\", \"county\": \"городской округ Екатеринбург\",
15   | \"state\": \"Свердловская область\", \"postcode\": \"620041\", \"country\": \"РФ\", \"country_code\": \"ru\"}",
16   "display_name": "30, улица Вайнера, Юго-Западный, Ленинский район, Екатеринбург, городской округ Екатеринбург",
17   | Свердловская область, Уральский федеральный округ, 620041, РФ",
18   "coords": "{\"lat\": \"56.8323468\", \"lon\": \"60.5958021064869\"}",
19   "shape": "[[60.595619, 56.8323957], [60.595674, 56.832237], [60.5959947, 56.8322703], [60.5959852, 56.8322979],
20   | [60.595619, 56.8323957]]",
21   "status": 1,
22   "rate_final": "6.12"
23 }

```

Рисунок 4.8 – Ответ к запросу на получение информации о конкретном здании

#### 4.1.2 Экранные формы

Пользовательский интерфейс прототипа не содержит элементы доступа к функциям системы, предоставляемым API. В качестве альтернативы предлагаемое решение включает в себя реализацию пользовательского web-интерфейса САТУ, работающего как клиентское приложение, использующее API.

Доступ к web-приложению осуществляется через браузер. Доступ к основным страницам осуществляется с помощью главного меню, расположенного в верхней части каждой web-страницы. Главная страница (рисунок 4.9) поделена на две части: панель поиска зданий и интерактивная карта Google Maps. Здания, которые были проанализированы САТУ, выделяются цветом по шкале от красного до зелёного в соответствии с оценкой энергоэффективности.

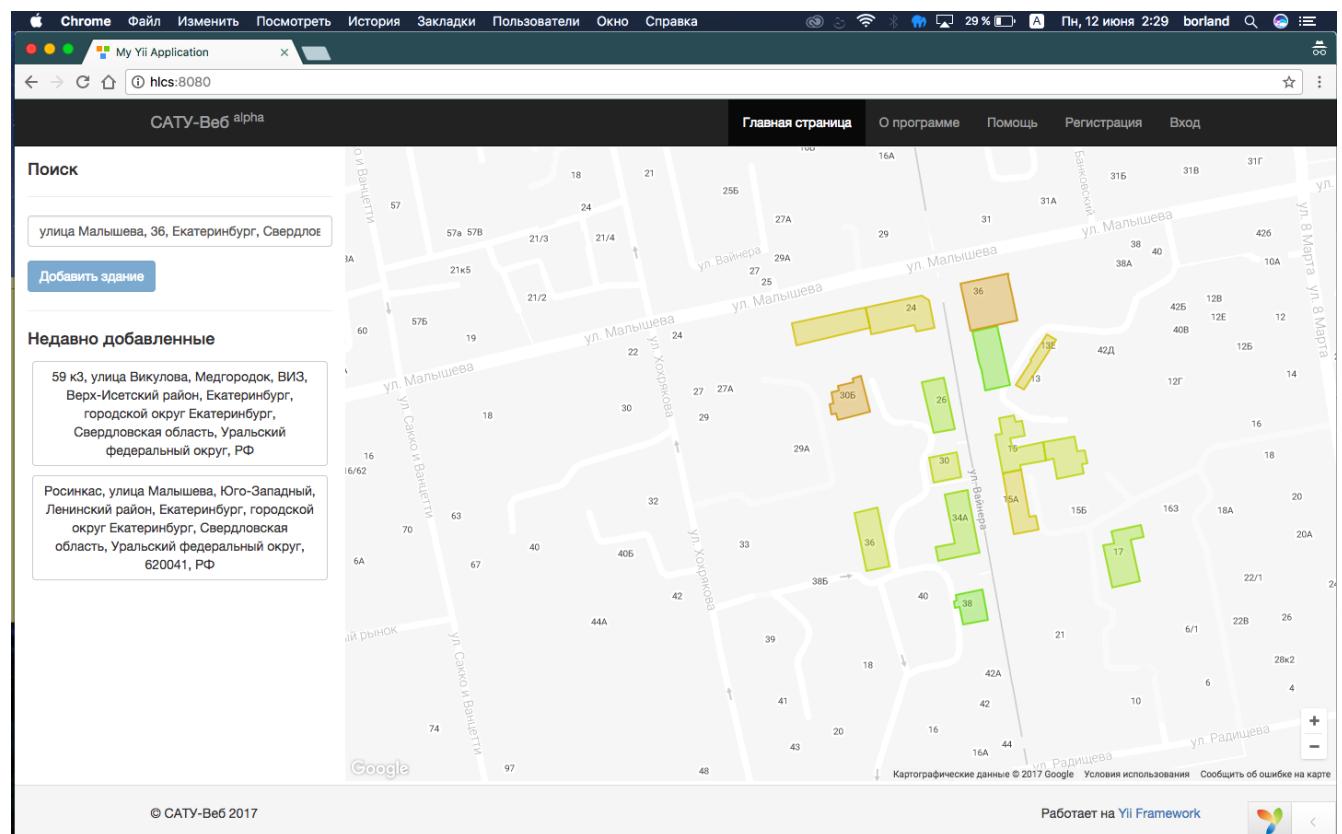


Рисунок 4.9 – Главная страница веб-интерфейса САТУ

При введении пользователем интересующего адреса в поле поиска формируется выпадающий список (рисунок 4.10) с доступными в ГИС адресами зданий.

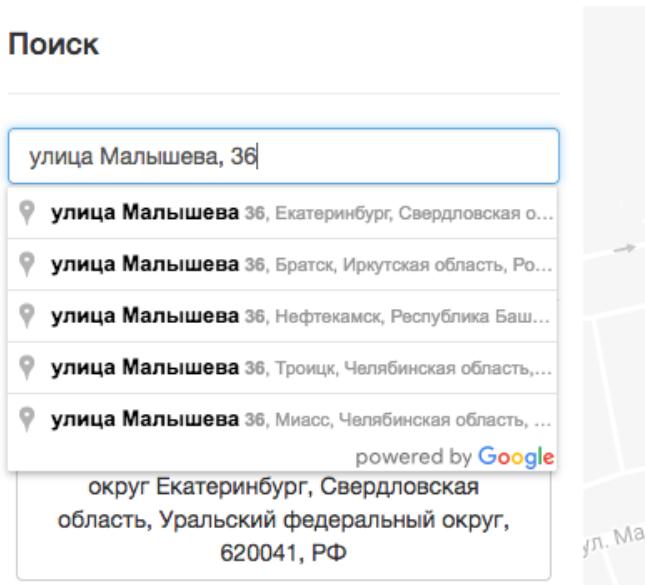


Рисунок 4.10 – Процедура поиска здания в ГИС Google Maps

После выбора адреса из списка возможно два варианта поведения: если здание найдено в САТУ, оно отобразится в центре карты; в противном случае здание дополнительно выделяется розовым цветом (рисунок 4.11).

При нажатии левой кнопкой мыши по подсвеченному зданию на карте рядом с ним отображается всплывающее окно с краткой информацией и ссылкой на страницу с подробными сведениями (рисунок 4.12).

В случае, если здание выделено розовым цветом, в панели поиска становится доступна кнопка “Добавить здание”. При нажатии на неё пользователь перейдёт на страницу “Добавить/изменить здание” (рисунок 4.13). Эта страница позволяет просмотреть историю съёмки здания. При необходимости пользователь может добавить новую серию снимков, нажав на кнопку “Добавить серию снимков”.

Загрузка серии снимков через web-приложение в браузере проходит в два этапа:

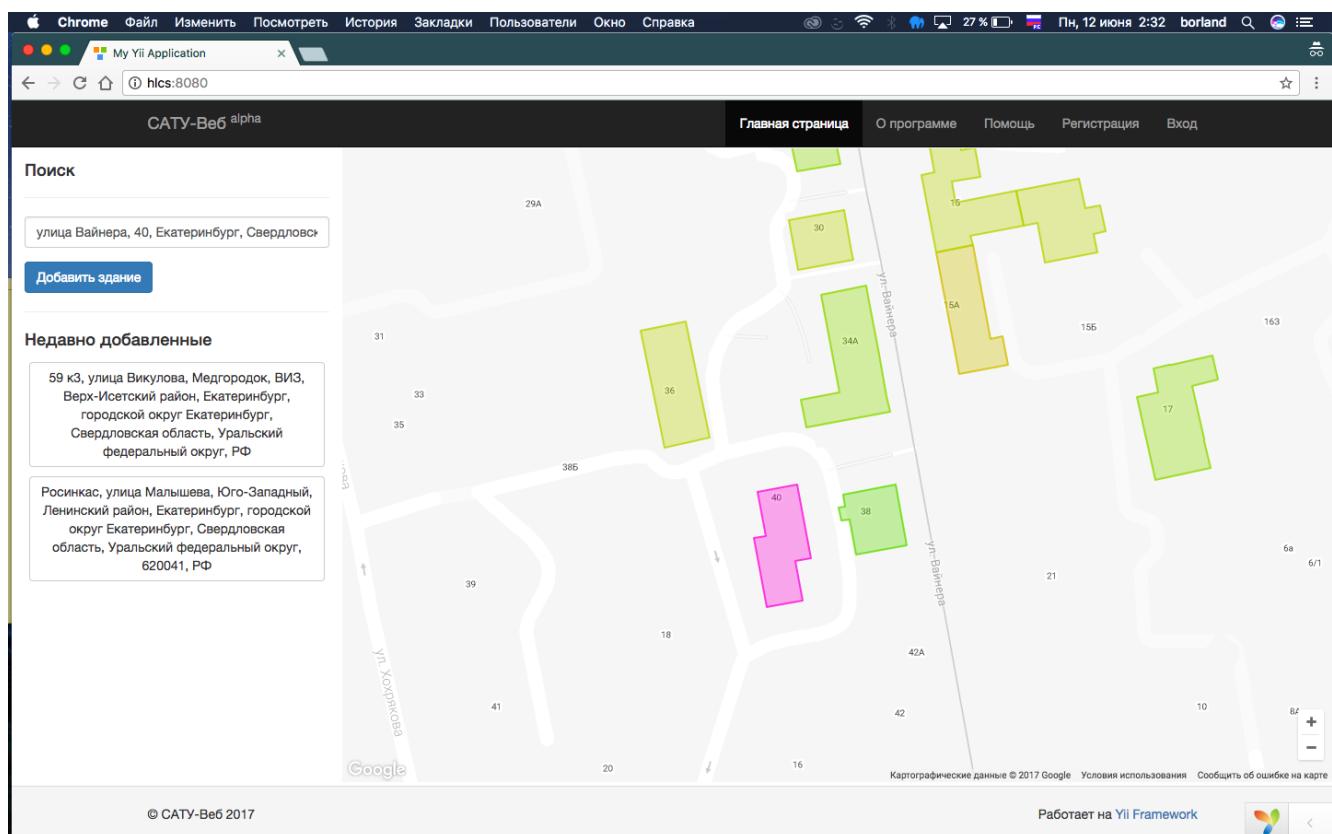


Рисунок 4.11 – Отображение здания, не обнаруженного в САТУ

- выбор типа, устройства съёмки, непосредственно загрузка ИК изображений (рисунок 4.14)
- указание дополнительных данных о загруженных снимках (положение камеры) (рисунок 4.15)

Форма загрузки изображений использует два способа выбора файлов: перетаскивание в область формы вручную; указание каталога, содержащего файл с помощью всплывающего при нажатии на кнопку “Выбрать” окна. После выбора в форме отображаются пиктограммы изображений и указывается их размер. При желании изображения можно удалить из формы.

После успешного добавления здания в САТУ на главной странице в разделе “Недавно добавленные” отображаются последние зарегистрированные системой здания. При нажатии на кнопку с адресом здания карта мгновенно переносит обзор на него.

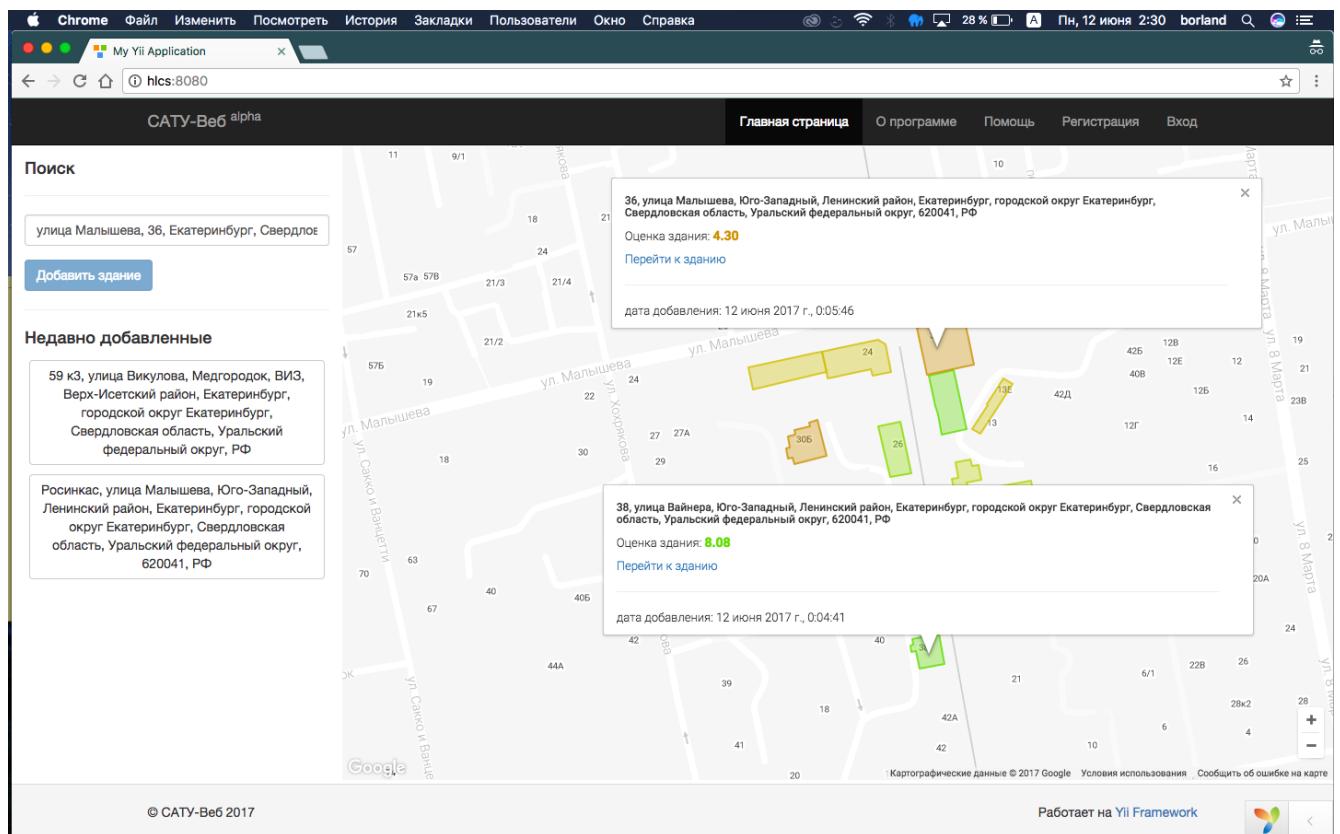


Рисунок 4.12 – Взаимодействие с проанализированными зданиями на карте:  
отображение всплывающих окон

Некоторые действия, такие как загрузка снимков и добавление зданий, недоступны пользователям, не представившимся в системе. Для этого предусмотрены страницы с формами регистрации и входа в систему (рисунки 4.16, 4.17).

## 4.2 Результаты и выводы

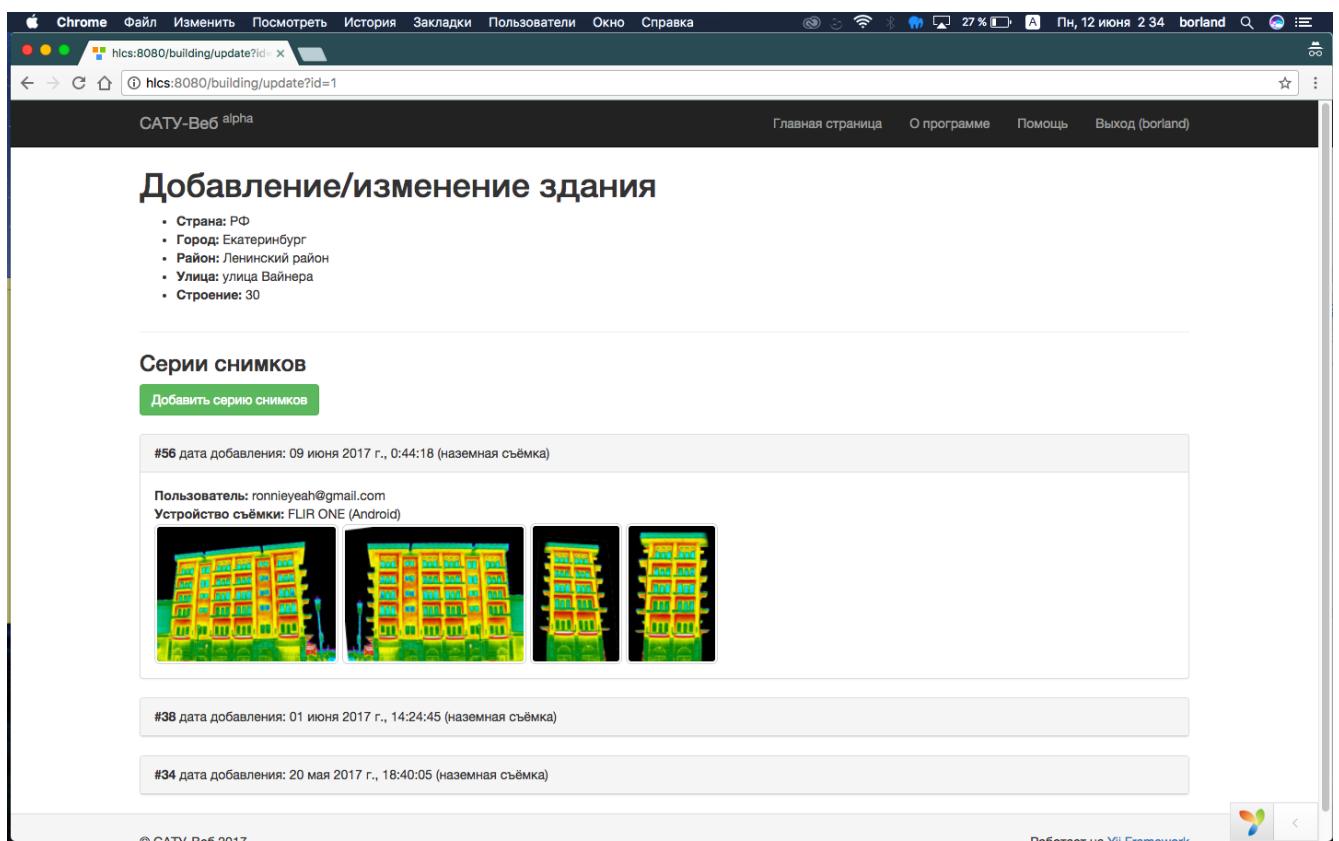


Рисунок 4.13 – Форма добавления/изменения здания

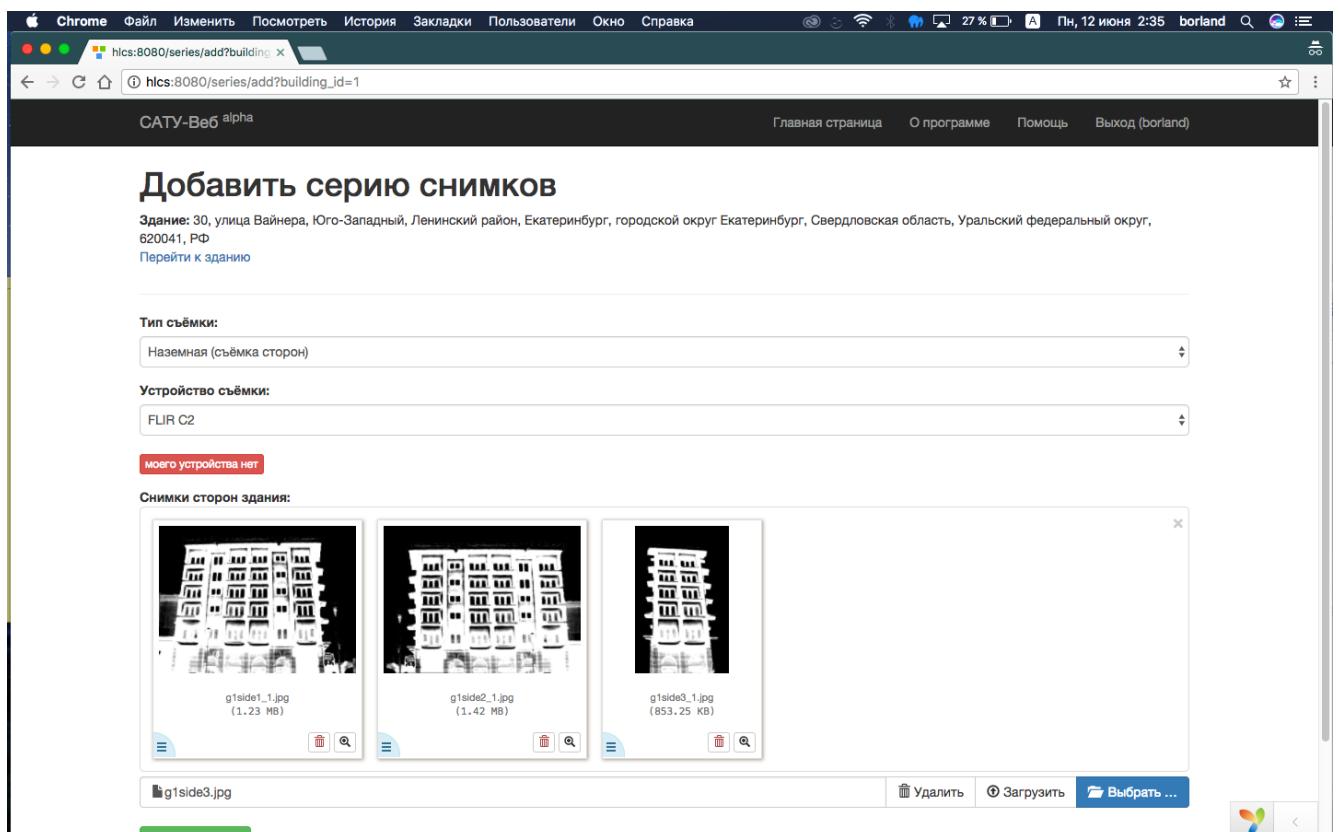


Рисунок 4.14 – Форма добавления серии снимков (этап 1)

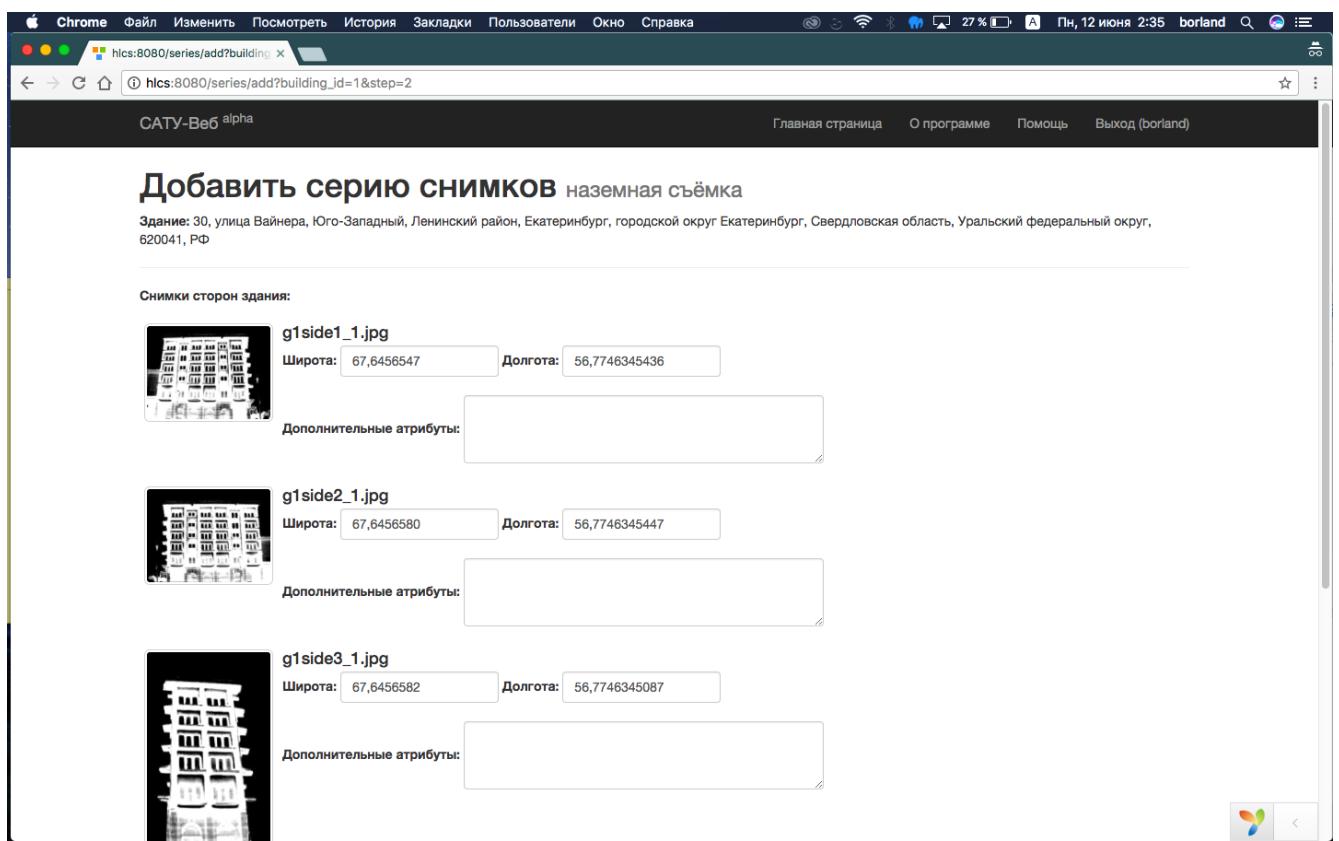


Рисунок 4.15 – Форма добавления серии снимков (этап 2)

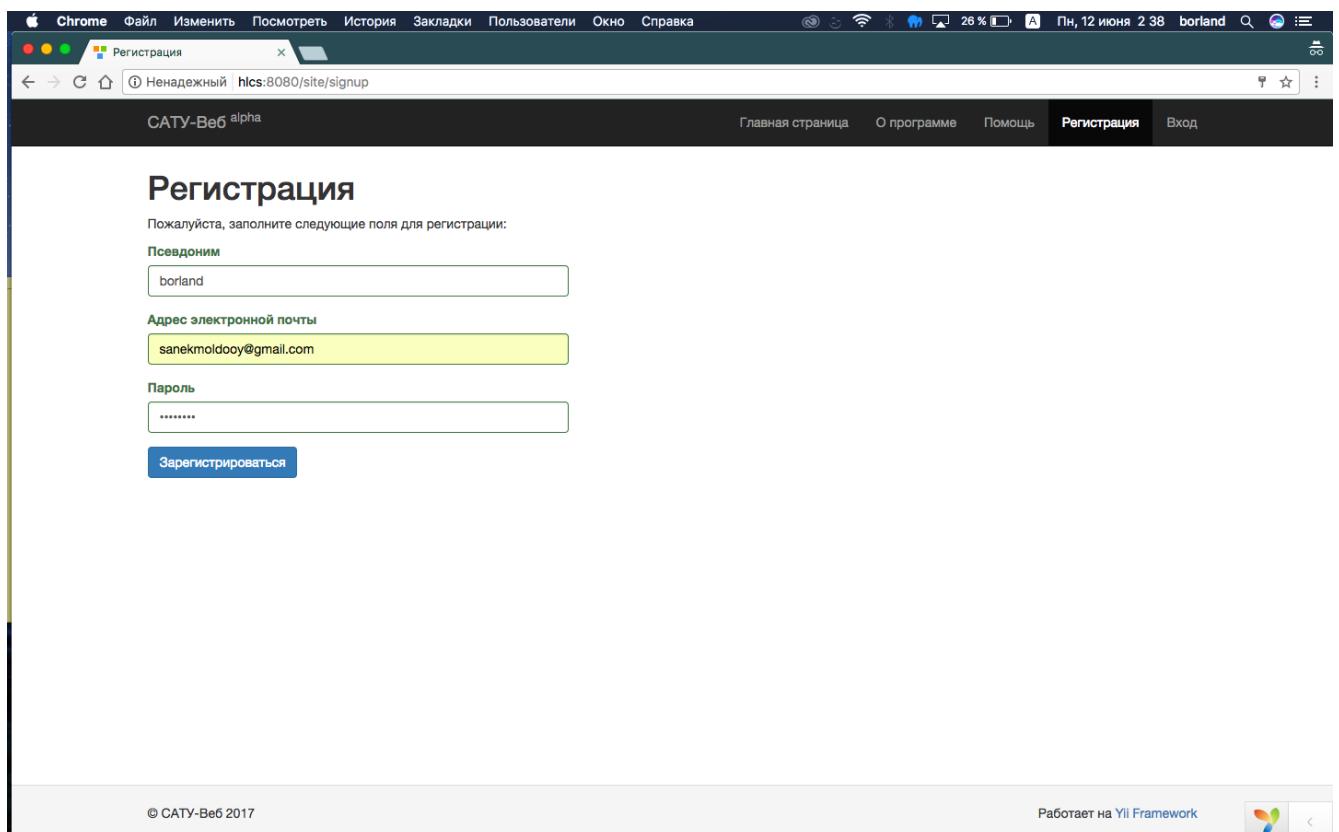


Рисунок 4.16 – Форма добавления серии снимков (этап 1)

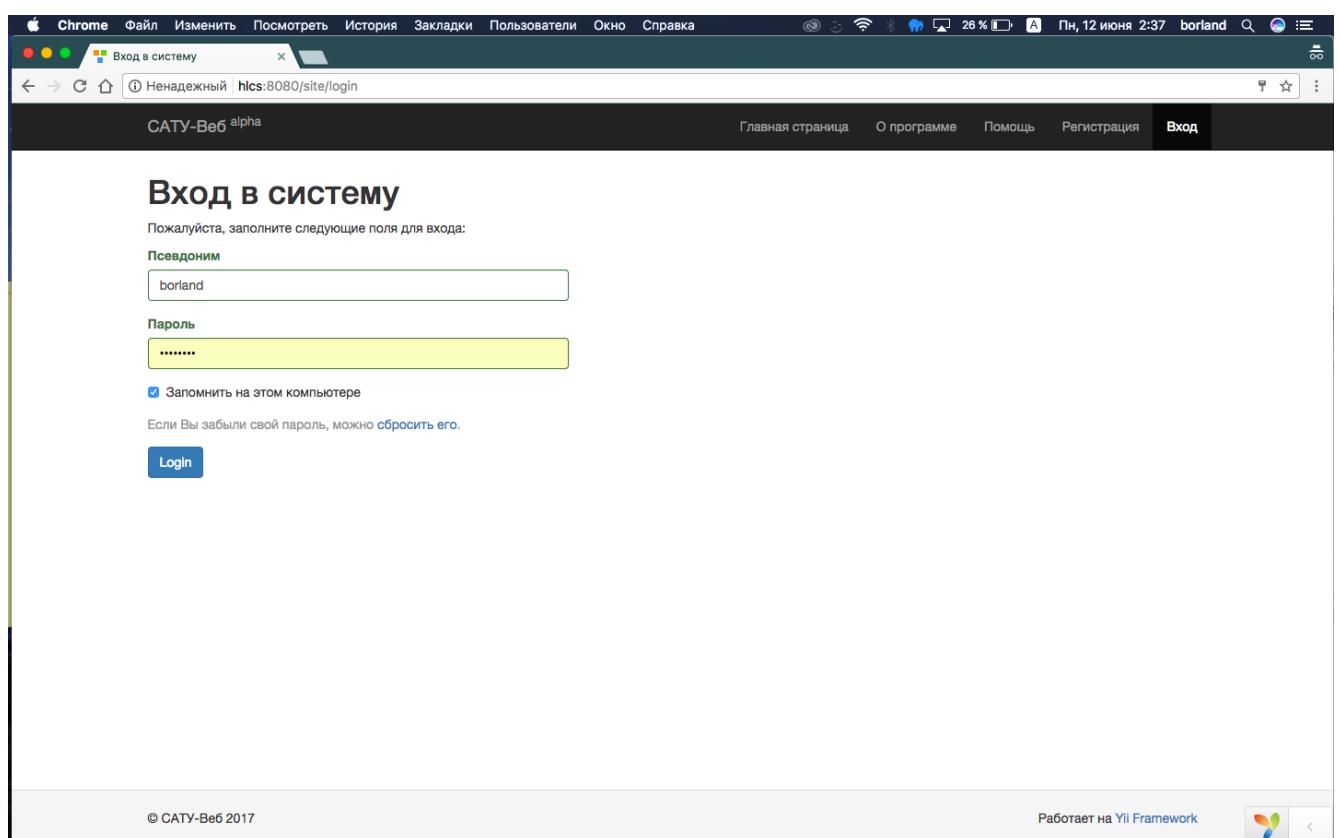


Рисунок 4.17 – Форма добавления серии снимков (этап 2)

## **Список использованных источников**

1. *Анализ потребления тепловой энергии на отопление многоквартирных домов как способ повышения энергоэффективности в сфере ЖКХ* [Электронный ресурс] / Дирекция по проблемам ЖКХ // Аналитический центр при Правительстве Российской Федерации. — 2013. — Режим доступа: [http://gkh-altay.ru/d/205499/d/06\\_24\\_kr\\_stol\\_analitika\\_dor\\_abotannaya\\_po\\_rezultata\\_m.pdf](http://gkh-altay.ru/d/205499/d/06_24_kr_stol_analitika_dor_abotannaya_po_rezultata_m.pdf) (дата обращения: 29.03.2017).
2. *Heating and cooling energy trends and drivers in buildings* [Текст] / Ürge-Vorsatz D., Cabeza L. F., Serrano S., Barreneche C., Petrichenko K. // Renewable and Sustainable Energy Reviews — Budapest, Hungary: Elsevier, 2015. — № 41 — С. 85-98.
3. *What are the sources of home heat loss?* [Электронный ресурс] / Wilson L. // Shrink That Footprint — Режим доступа: <http://shrinkthatfootprint.com/home-heat-loss> (дата обращения: 29.03.2017).
4. *Detecting sources of heat loss in residential buildings from infrared imaging* [Электронный ресурс] / Chen S., Chen E. // Massachusetts Institute of Technology. Dept. of Mechanical Engineering — 2011. — Режим доступа: <http://hdl.handle.net/1721.1/68921> (дата обращения: 20.02.2017).
5. *Energy use in buildings in a long-term perspective* [Текст] / Ürge-Vorsatz D., Petrichenko K., Staniec M., Eom J. // Current Opinion in Environmental Sustainability — Budapest, Hungary: Elsevier, 2013. — № 5 — С. 141-151.

# **Приложение А**

## **Функционально-структурные модели**

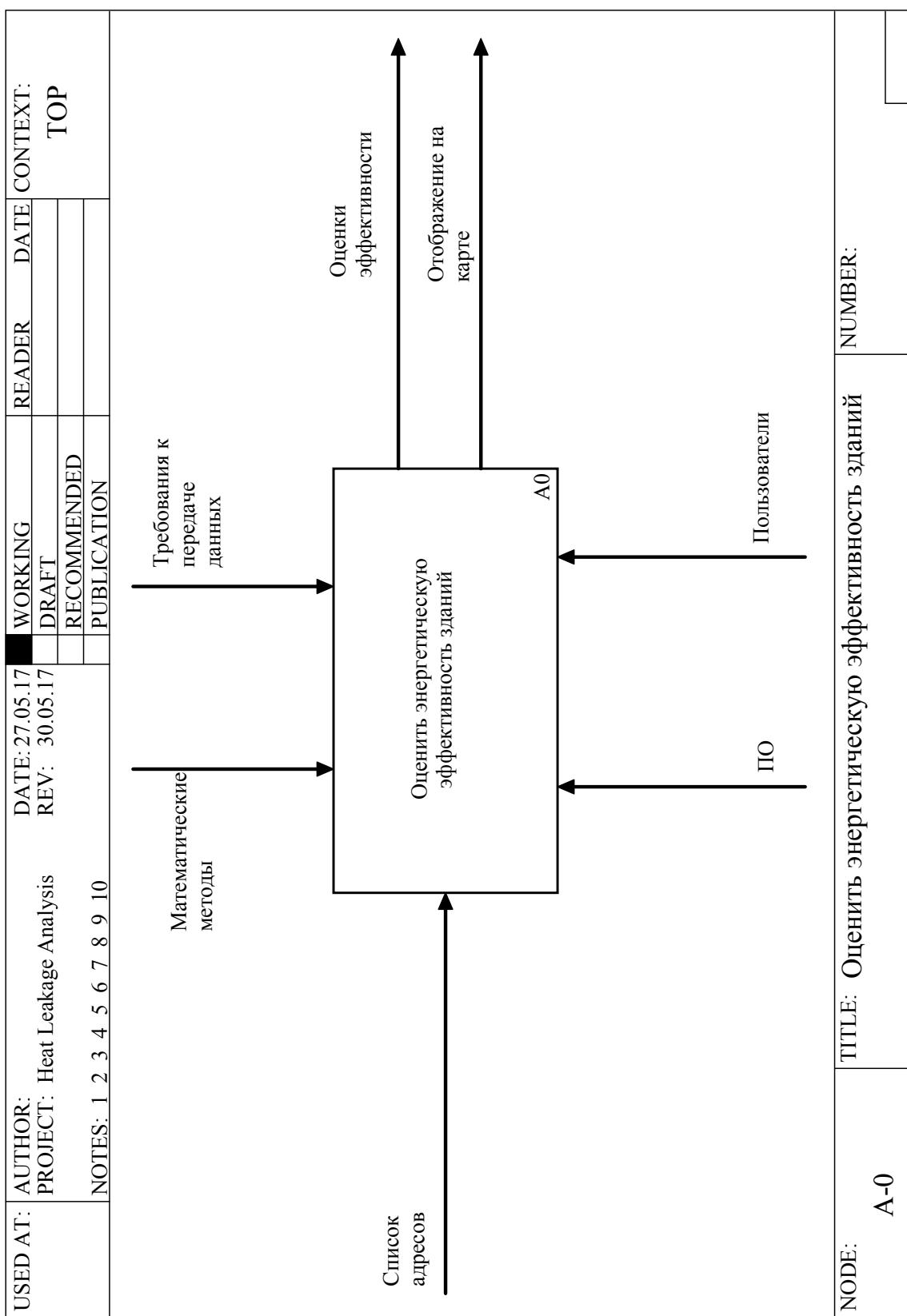


Рисунок А.1 – Контекстная диаграмма функционально-структурной модели предлагаемого решения

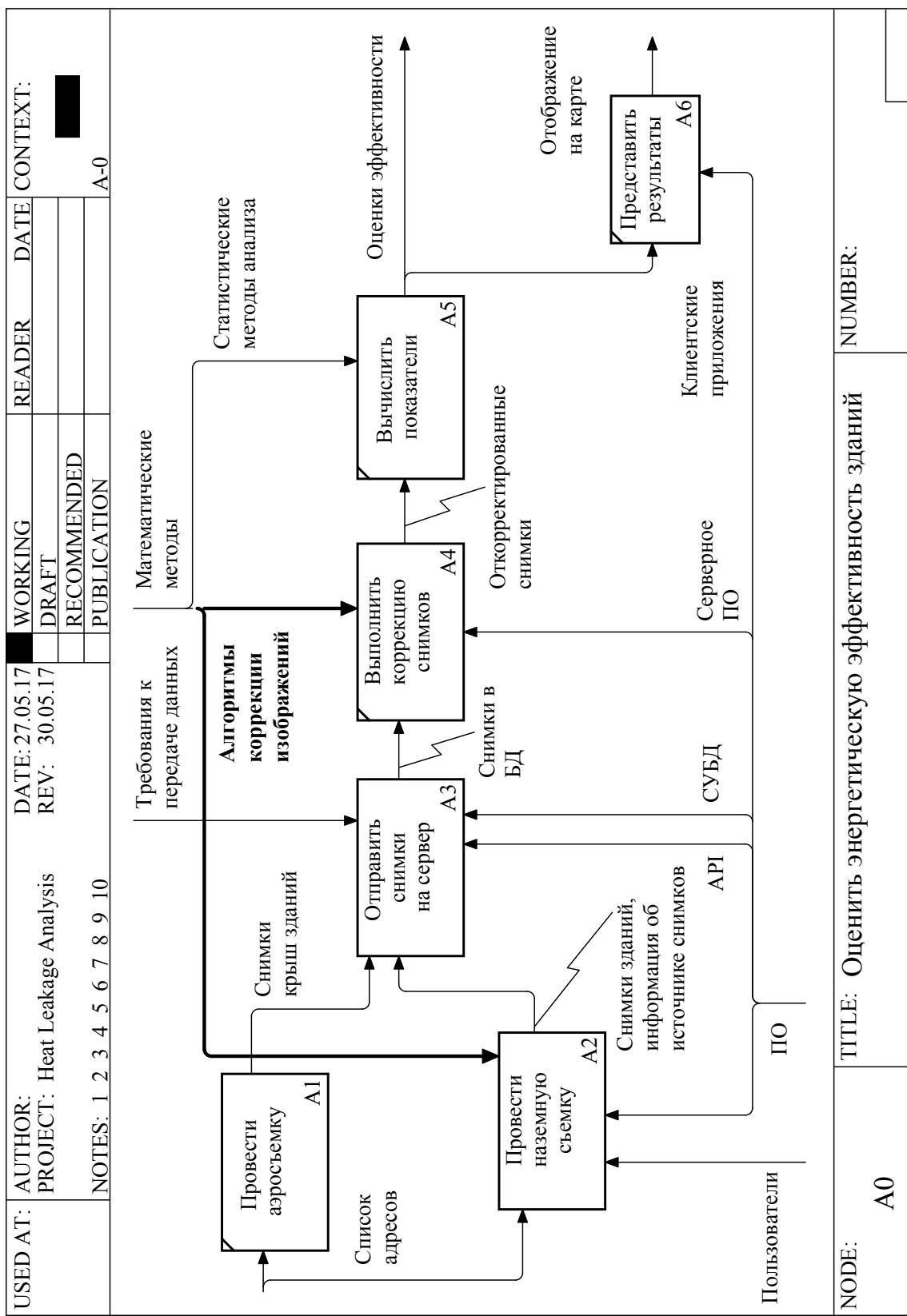


Рисунок А.2 – Диаграмма декомпозиции 1 уровня функционально-структурной модели предлагаемого решения

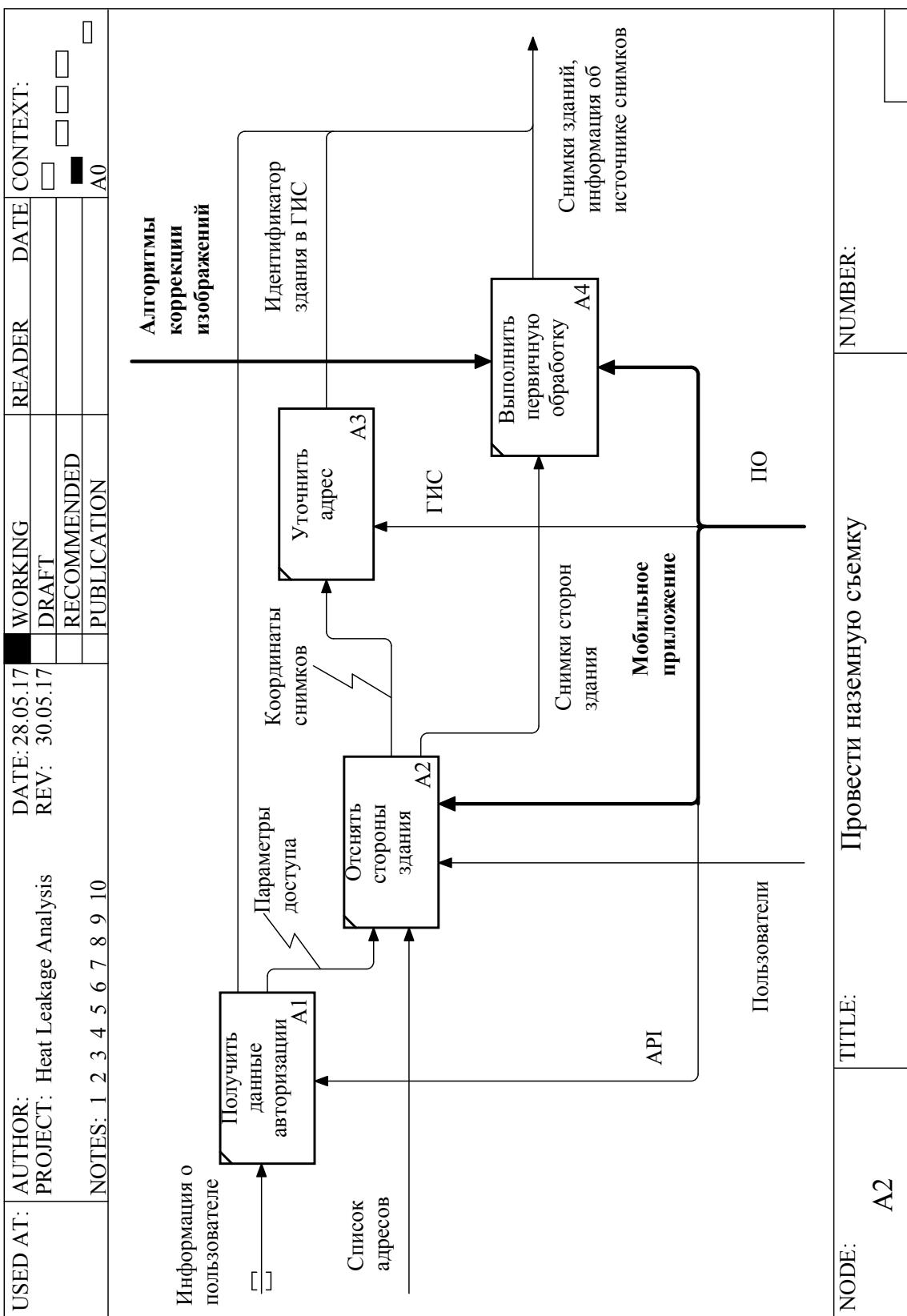


Рисунок А.3 – Диаграмма декомпозиции блока А3 функционально-структурной модели предлагаемого решения

# **Приложение Б**

## **Техническое задание**