

**КУРСОВОЙ ПРОЕКТ**

**По дисциплине: МДК 01.01 Разработка программных модулей**

**Тема: Разработка модуля «Работа с участниками» приложения «Молодежный форум»**

**Специальность 09.02.07 «Информационные системы и программирование»**

|  |  |  |
| --- | --- | --- |
| **Выполнила студентка группы 309ИС-22** | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Е.С. Зайцева** |
| **Руководитель** | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Л.Б. Гусятинер** |

**Москва 2024**

**ВВЕДЕНИЕ**

Тема курсового проекта: Разработка программного модуля «Работа с участниками» для приложения «Молодежный форум».

Актуальность данной темы курсового проекта обусловлена динамикой автоматизации процессов управления в сфере организации мероприятий в целом и процессом работы с участниками мероприятия, в частности.

В современном мире, где значительная часть коммуникации, особенно после пандемии COVID-19, осуществляется посредством сети Интернет, сложно переоценить важность и эффективность живого общения между людьми. Поэтому многие организации проводят очные встречи для обмена опытом, знаниями и практиками.

При подготовке и проведении подобных мероприятий необходимо учитывать множество процессов, ведь впечатление от посещения события, напрямую влияет на имидж компании.

Для форумов, относящихся к категории «молодежный», особенно важно наличие инновационных технологий в процессе участия в мероприятии. Молодежь с меньшим интересом относится к событию, если на нем отсутствуют программные решения, связанные с «будущим». Повышающим лояльность участников фактором являются новые креативные способы выполнения задач.

Основываясь на опыте организации очных форумов различных проектов, можно выделить важный аспект: чем более чётко обозначены обязанности каждого ответственного, тем более качественным будет результат.

Однако, не всегда представляется возможным найти необходимое количество людей, для реализации мероприятия. Преодоление этой проблемы затрудняется ограниченным бюджетом и недостаточным количеством времени для обучения помощников.

Решением может стать применение информационных технологий, которые способны автоматизировать часть процессов, что позволит сократить состав штаба организаторов и время, затрачиваемое на подготовку.

Это выгодно проектам, которые стремятся проводить более масштабные мероприятия с регулярной частотой или с меньшими экономическими затратами.

***Цель выполнения проекта*** – разработка программного модуля приложения для автоматизации молодежного мероприятия.

***Объект исследования*** – организация мероприятий

***Предмет исследования*** – процесс работы с участниками на молодежном форуме

Для достижения поставленной цели курсового проекта необходимо решить следующие ***задачи:***

* Проанализировать предметную область и обосновать актуальность проекта.
* Изучить существующие программные решения и выделить процессы, которые необходимо автоматизировать.
* Выбрать язык программирования, среду разработки и ЖЦ ПО.
* Спроектировать общую структуру программного продукта.
* Продумать необходимые функции для работы программы.
* Разработать программное обеспечение.
* Провести отладку и тестирование программного обеспечения.
* Разработать программную документацию для сопровождения программного обеспечения.
* Подготовить презентацию проекта для защиты перед потенциальными инвесторами или покупателями.

1. **АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ОБОСНОВАНИЕ АКТУАЛЬНОСТИ ПРОЕКТА**
   1. **Анализ предметной области и обоснование актуальности**

В данном курсовом проекте предметной областью для решения поставленных задач является (организация) молодежный форум.

***Форум*** — это мероприятие, на котором люди собираются вместе для обсуждения определенной темы или проблемы. [1] А также обмена знаниями, опытом и идеями, образования деловых связей, поиска инвесторов, обретения полезных знакомств или получения гранта на реализацию проекта.

Данного типа мероприятия проводятся повсюду: в больших корпорациях, в государственных органах власти, в школах и вузах; среди работников медицины, культуры, педагогики и бизнеса. Тематика и масштаб события может быть абсолютно разным.

К числу известных форумов относятся Петербургский международный экономический форум (ПМЭФ), Всероссийский молодежный форум «Машук» и др. Также в рамках деятельности нашего колледжа, студенты и администрация колледжа регулярно устраивают различные форумы (II Межрегиональный форум «Предпринимательство: новые идеи молодежи»; Форум «Волонтерство в СПО» и др.).

***Основные посетители*** молодежного форума – это люди в возрасте от 14 до 35 лет, каждый из которых неравнодушно относится к теме форума, хочет узнать что-то новое или поделится своими знаниями.

«Проведение масштабных «форумных кампаний» в России началось сравнительно недавно, но они активно популяризируются в молодежной среде, развиваются старые и появляются новые площадки, которые ежегодно посещают тысячи молодых людей.

С 2015 г. постоянно растет количество участников форумов, что говорит об охвате все большего количества молодежи и профильной работы с ней. В 2015 г. в молодежных форумах России приняли участие около 170 000 чел., в 2016 г.— 360 000 чел., в 2017 г.— 500 000 чел., в 2018 г.— 800 000 чел., в 2019 г.— 900 000 чел.» [2] Наглядное представление тенденции роста количества участников представлено на диаграмме (Рис.1.1)

Рисунок 1.1 Динамика количества участников молодежной «форумной кампании»

Рост количества участников свидетельствует о необходимости внедрения новых решений в существующую организационную структуру, для поддержания высокого уровня обслуживания при увеличении числа посетителей.

У каждого форума есть собственный стиль или брендбук, который регламентируют дизайн афиш, мерча и помещения, поэтому важно учитывать пожелания заказчиков при создании шаблонов. Это влияет на уникальность и узнаваемость проекта.

Изучение существующих решений позволило найти схожую с планируемым функционалом программу REGPART, команда которой с 2013 года занимается организацией регистрации участников различных мероприятий. [4]

* 1. **Процессы, представляемые к автоматизации**

Организация мероприятия представляет собой многоступенчатый процесс, требующий от дирекции комплексного подхода и координации следующих аспектов:

* Взаимодействие с поставщиками – обеспечение закупок надлежащего качества и исполнения установленных сроков.
* Закупка и выдача мерча – разработка дизайна, обеспечение достаточного количества продукции для всех участников.
* Организация места проведения форума – поиск и аренда подходящей площадки и необходимого оборудования.
* Составление программы мероприятия – составление детального плана, включающего все запланированные сессии, выступления и активности.
* Приглашение и согласование выступающих спикеров.
* Управление регистрацией участников (при наличии конкурса на участие, также – отбор поступающих заявок).

Возникает необходимость в компьютерной обработке данных поступающих от разных категорий участников и организаторов, с целью централизованного хранения информации и дальнейшего анализа эффективности мероприятия.

***Управление регистрацией участников*** — это комплексный процесс, включающий в себя все действия, необходимые для эффективного сбора, обработки и управления информацией об участниках мероприятия.

Варианты регистрации и пропуска участников на событие:

1. *Фактически свободный вход* — когда участники формально проходят через некую входную группу, но их проверяет только служба безопасности. Могут возникнуть недопонимания с партнерами, из-за отсутствия системы отслеживания количества посетителей.
2. *Проход по билетам через волонтеров* — когда на входе стоят члены команды и участники предъявляют им билеты (распечатанные или в электронном виде).
3. *Получение бейджа из заранее напечатанных.* На основании данных из своей базы организаторы печатают бейджи или подписывают их вручную. На стойке регистрации участник называет свою фамилию, по ней в стопках находят его бейдж, и на основании этого он проходит на мероприятие. Однако, при таком варианте регистрации, кто-то может назвать чужую фамилию и пройти на мероприятие.
4. *Печать бейджа на месте в обмен на предъявление документа, удостоверяющего личность*. Волонтер сверяется со списками, загруженными в компьютер на стойке регистрации, и, найдя нужную фамилию, печатает бейдж.
5. *Самостоятельная регистрация* — сервис, где участник прикладывает штрихкод к считывателю или вводит цифровой код из билета и получает бейдж, распечатанный волонтером или автоматически.
6. *Регистрация через приложение* — участник скачивает специально созданное для мероприятия приложение и после покупки билета проходит в нем регистрацию. На мероприятии через него он может распечатать бейдж. [5]

Последние два варианта самые предпочтительные, но при этом дорогостоящие, позволить их могут не все организации, что ещё раз показывает актуальность создания программного модуля, способного решить эту проблему и повышающего количество компаний, использующих инновационные технологии.

***Разрабатываемый программный модуль должен обеспечивать, реализацию следующих процессов:***

1. *Регистрация:* сбор и анализ личных данных участника необходимых для подтверждения личности и формирования итогового отчета о посещении мероприятия (Фамилия, имя и отчество, дата рождения, половая принадлежность, номер телефона, имя пользователя в социальной сети «Telegram»)
2. *Подтверждение финальной явки на мероприятие и отслеживание текущего количества участников*
3. *Визуализация информации об участниках:* этот процесс важен не только для составления отчета, но и для формирования представления о количестве участников разных категорий. Необходимо иметь возможность реализовать передачу этой информации другим организаторам или, в последствии, другим модулям программы. (Получение статистики обязательно, если на форуме предусмотрена выдача индивидуального комплекта мерча для каждого участника.)
4. *Выдача индивидуального бейджа:* наличие бейджа поможет людям правильно обращаться друг к другу, понимать роль человека на мероприятии, а также подтвердить свое участие, при выходе с площадки проведения форума.

Более подробно все процессы представлены на DFD – диаграмме потоков (Приложение 1, Рис.1.2)

***Организация входных и выходных данных:***

Исходные данные поступают в систему с помощью ввода пользователем данных о пришедших участниках или из заранее составленного списка зарегистрированных, тогда в обязанности пользователя будет входить проставление настоящего статуса участника каждого зарегистрированного. После анализа и обработки поступившей информации они сохраняются в системе классов и файлах с соответствующими названиями.

Выходными данными являются отчет с информацией участников, список заготовленных бейджей в текстовом формате и непосредственно бейдж зарегистрированного участника.

* 1. **Выбор и обоснование языка программирования, среды разработки и жизненного цикла ПО**

Для разработки программного модуля «Регистрация участников» выбран язык программирования Python и интегрированная среда разработки (IDE) PyCharm.

Выбор языка программирования Python обусловлен следующими факторами:

* *Высокая читаемость и простота:* Python отличается ясным и лаконичным синтаксисом, что упрощает разработку, отладку и поддержку кода. Это особенно важно для командной работы и дальнейшего развития модуля, как часть приложения.
* *Богатый набор библиотек*: это особенно актуально для реализации функциональности регистрации участников, которая может включать в себя взаимодействие с различными системами.
* *Большое сообщество разработчиков:* сообщество Python-разработчиков обеспечивает доступ к обширной документации, поддержке и готовым решениям

Выбор среды разработки PyCharm:

* *Удобство использования*: PyCharm имеет интуитивно понятный интерфейс, что повышает производительность разработчиков. Например, среда разработки сама подскажет имя переменной и каким типом данных она является
* *Инструменты для работы с данными:* PyCharm интегрируется с различными системами управления базами данных (СУБД), что в последствии упростит работу с данными участников.

Для разрабатываемого программного модуля «работа с участниками» подходит ***V-образная модель жизненного цикла***, которая характеризуется параллельным выполнением этапов разработки и тестирования.

Эта модель обеспечивает строгую последовательность и верификацию каждого этапа разработки, что полезно при работе с ограниченным количеством разработчиков и четко обозначенными требованиями.

В рамках V-образной модели, каждый этап разработки имеет соответствующий этап тестирования. Только после успешного прохождения тестирования на каждом этапе, проект переходит к следующему.

V-образная модель обладает рядом преимуществ:

* *Раннее выявление ошибок:* благодаря параллельному тестированию, ошибки обнаруживаются на ранних этапах жизненного цикла, что снижает время исправления.
* *Повышенное качество ПО:* Строгая последовательность и верификация каждого этапа гарантируют высокое качество конечного продукта.
* *Четкая документация:* V-образная модель требует тщательной документации на каждом этапе, что облегчает сопровождение и дальнейшее развитие ПО.
* *Лучшее понимание требований:* Процесс верификации требований на ранних этапах обеспечивает более точное понимание потребностей заказчика.

V-образная модель обеспечивает более предсказуемый процесс разработки и гарантирует высокое качество конечного продукта. Поэтому с уверенностью можно сказать, что данная модель жизненного цикла программного продукта соответствует жизненному циклу программного модуля «Работа с участниками».

**2. ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ ПРОГРАММНОГО МОДУЛЯ**

**2.1. Проектирование общей структуры программного продукта**

В процессе проектирования основное внимание уделяется логическому решению, обеспечивающему выполнение основных требований.

Рассмотрим построение ***системы классов***, где будет хранится информация о зарегистрированных участниках, категориях участия и мероприятиях, организованных пользователем. Информация будет использоваться дирекцией и другими организаторами молодежного форума, а также его посетителями.

Выделим следующие классы, их элементы и методы:

* Участник, основной класс используемый как шаблон для заполнения отчета по каждому посетителю: ID, фамилия, имя, отчество, дата рождения, пол, телефон, имя пользователя в телеграм (для связи и отображения на бейдже), статус присутствия (для мониторинга присутствующих участников)
* Каждый участник будет принадлежать к определенной категории. Класс Категории состоит из ID, названия категории и списка участников, которые ей принадлежат
* Мероприятие, класс от которого будут зависеть другие классы. Необходим для функции, осуществляющую доступ для работы с несколькими мероприятиями, как отдельными проектами.
* Бейдж: вспомогательный класс, в котором хранится необходимая для генерации бейджей информация. Зависим от трех других классов

Анализ объектов позволяет выделить, основные классы: Участник, Категории, Мероприятие, которые могут стать основополагающими при создании базы данных для программного модуля приложения.

Связи, методы и атрибуты классов представлены на Диаграмме классов (Приложение 1, рис.2.1)

Для создания базы данных, необходимо будет воспользоваться ER – диаграммой (Приложение 1, рис.2.2)

* 1. **. Основные функции и компоненты программного продукта**

|  |  |  |  |
| --- | --- | --- | --- |
| *№* | *Название* | *Функция* | *Класс* |
| 1 | change\_status | Изменение статуса участника. | Uchastnik |
| 2 | add\_uch | Добавление участника в категорию. С помощью объявленного объекта в программе. | Category |
| 3 | add\_uch\_vvod | Добавление участника в категорию. Пользовательский ввод. | Category |
| 4 | add\_cat | Добавление категории в мероприятие. Пользовательский ввод. | Meropriatie |
| 5 | add\_all\_cat | Добавление сразу всех категорий в мероприятие. Пользовательский ввод. | Meropriatie |
| 6 | Spisok\_cat | Запись в файл список категорий на мероприятии | Функциональная |
| 7 | clear\_Spisok\_cat | Очистка списка с категориями | Функциональная |
| 8 | Spisok | Запись в файл списка участников по категории | Функциональная |
| 9 | clear\_Spisok | Очистка списка участников | Функциональная |
| 10 | Generate\_badges | Генерация бейджей | Функциональная |
| 11 | clear\_Badges | Очистка файла с бейджами | Функциональная |
| 12 | New\_spisok | Обновление списка участников | Функциональная |
| 13 | New\_badges | Обновление списка бейджей | Функциональная |
| 14 | Badge\_w | Вывод индивидуального бейджа на печать | Функциональная |

Таблица 2.1 Список функций в программном модуле и их расположение

* 1. **Разработка программного продукта**

Для реализации проекта «Работа с участниками» приложения «Молодежный форум» будет использована PyCharm - интегрированная среда разработки на языке Python.

Основные функции, которые автоматизируют процессы при управлении регистрацией участников это Spisok, Generate\_badges и Badge\_w.

***Spisok:***

# функция записывает в файл список участников по категории

def Spisok(ct, filename="spisok\_text.txt"):

try:

with open(filename, "a", encoding="utf-8") as file:

file.write(f'{ct.name\_c}:\n')

for uchastnik in ct.people:

file.write(str(uchastnik) + "\n")

except Exception as fall:

print(f"Ошибка при записи в файл: {fall}")

***Generate\_badges:***

# функция для генерации бейджей  
def Generate\_badges(meropriatie, category):  
 with open('badges.txt', 'a', encoding='utf-8') as file: for people in category.people:  
 badge = Badge(people, meropriatie, category)  
 file.write(str(badge) + '\n')

Для работы функции Badge\_w необходимо установить библиотеку:

pip install docxtpl

А также прописать в коде программы импорт из этой библиотеки:

from docxtpl import DocxTemplate

***Badge\_w:***

# функция выводит бейдж участника в Word  
def Badge\_w(MR):  
 doc = DocxTemplate("бейдж.docx")  
 c = int(input('Генерация бейджа. Введите ID категории: ')) - 1  
 u = int(input('Генерация бейджа. Введите ID участника: ')) - 1

context = {'name\_m': MR.name\_m, 'cat\_mer': MR.cat\_mer,  
 'familia': MR.categories[c].people[u].familia,'ima': MR.categories[c].people[u].ima , 'teleg': MR.categories[c].people[u].teleg, 'name\_c':MR.categories[c].name\_c}  
 doc.render(context)  
 doc.save("бейдж-final.docx")

Чтобы эти функции выполнялись, необходимо было создать классы в который будут описаны все атрибуты для дальнейшей работы.

***2.3.1. Описание алгоритма выполнения программы***

- Создание нового мероприятия, как проекта, в котором будут выполняться остальные функции, храниться данные (Рис.2.3):



Рисунок 2.3

- Вводится количество категорий участников, которые будут задействованы на мероприятии (Рис.2.4):



Рисунок 2.4

- Вводится ID и названия всех категорий (Рис.2.5):

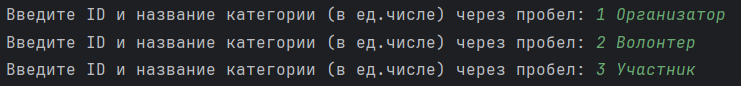


Рисунок 2.5

- Вводятся новые участники, которые пришли на мероприятие без предварительной регистрации (Рис.2.6):

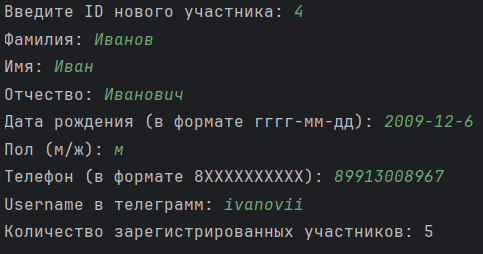


Рисунок 2.6

- Генерируется бейдж пришедшего участника для печати (Рис.2.7):

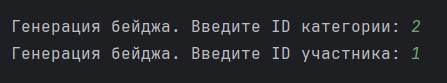


Рисунок 2.7

Таким образом, программа включает в себя необходимый функционал, анализ и обработку входных данных, формирование отчета

Проект поделен на 2 модуля:

Первый Kod\_2.py: состоит из классов и функций

Второй Itog.py: состоит из вызова этих функций

Работа второго модуля представлена в блок-схеме (Приложение 1, рис.2.8)

**ПРИЛОЖЕНИЕ 1**

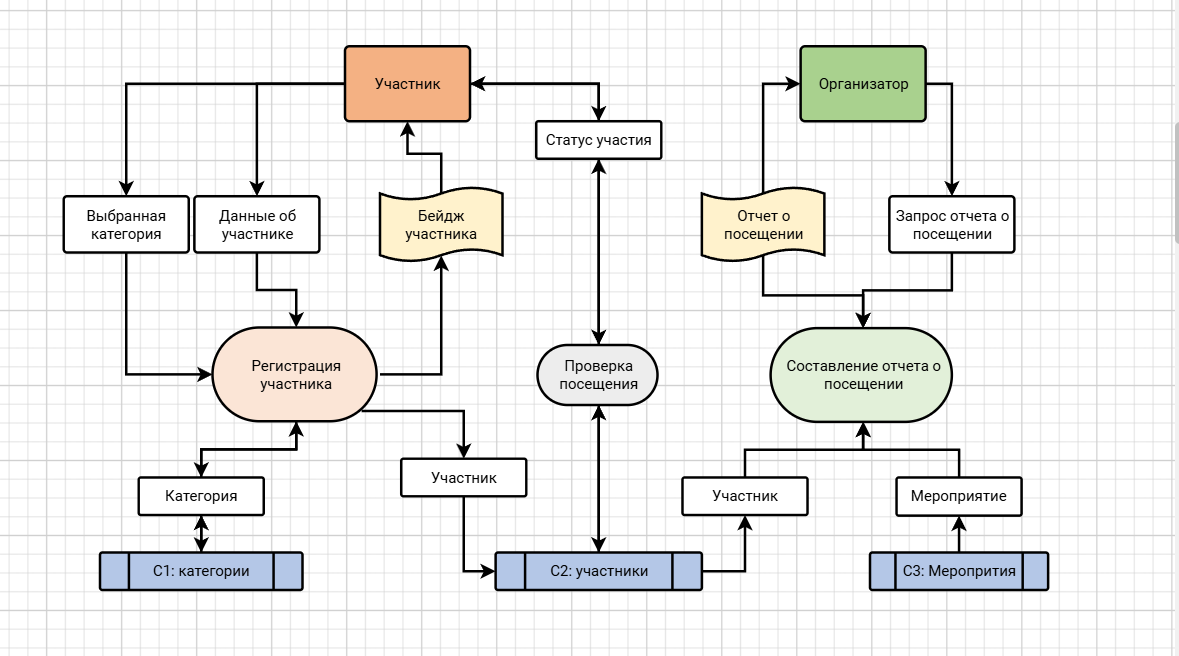
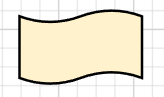


Рисунок 1.2 DFD Диаграмма потоков для модуля работа с участниками

Примечание 1 - схема составлена на основе, нотации Йордона-де Марко. Введено дополнительное обозначение  для более наглядного представления, которое означает итоговый выводимый документ.

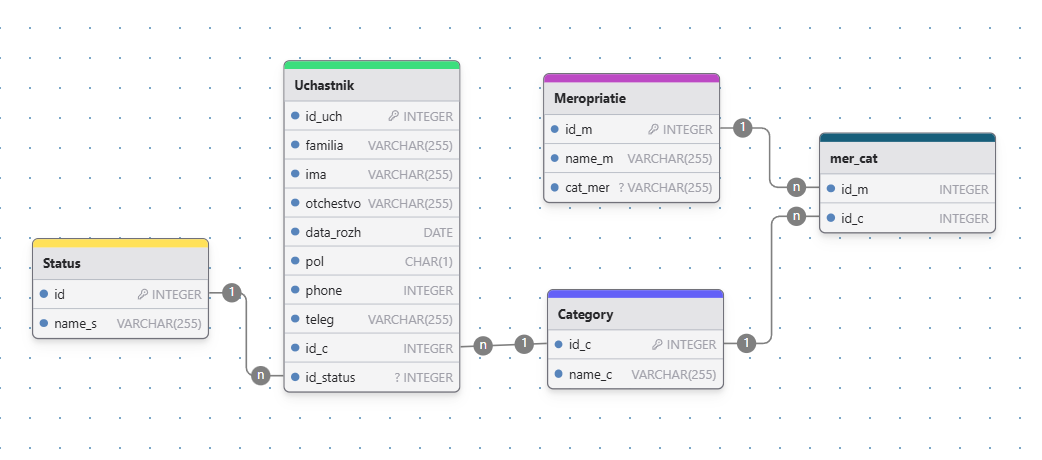


Рисунок 2.2 ER – модель базы данных программного модуля

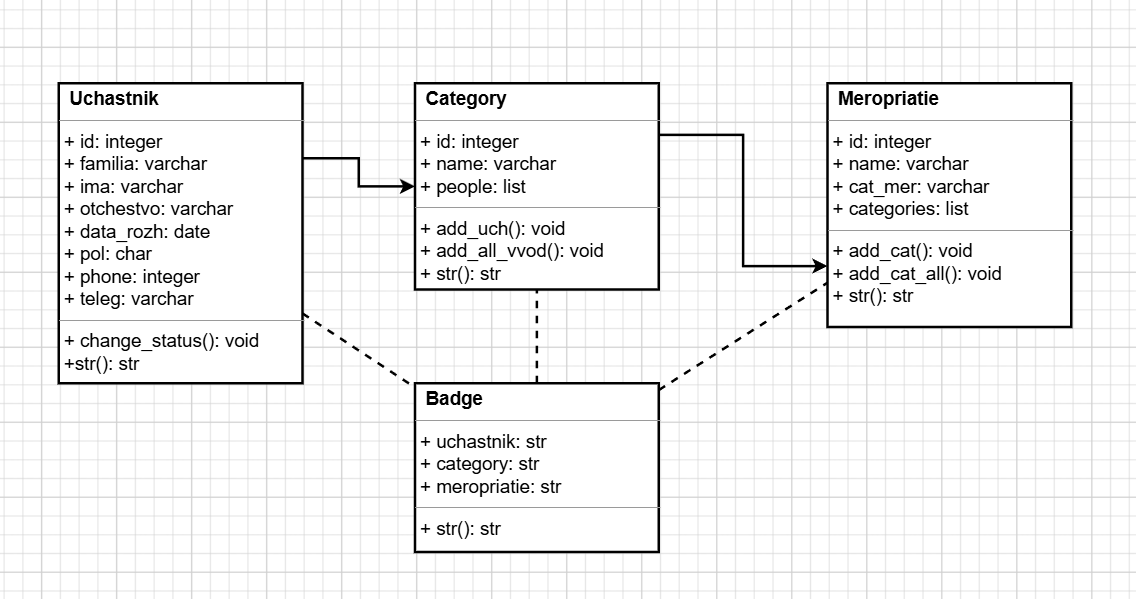


Рисунок 2.2 Диаграмма классов

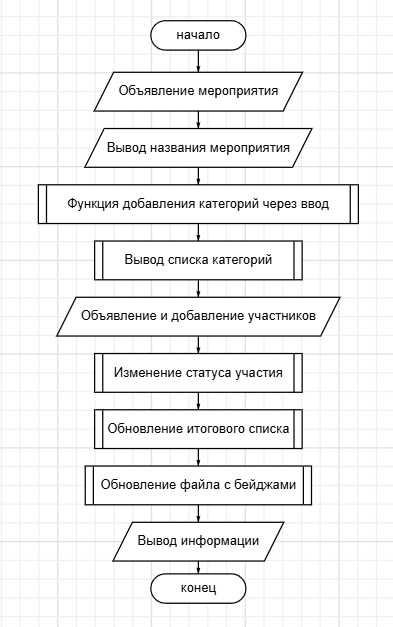


Рисунок 2.8 Блок – схема модуля Itog

**ПРИЛОЖЕНИЕ 2**

****

Примечание 2 – Пример сгенерированного на основе шаблона бейджа. Выполнен в фирменном стиле Департамента городского имущества города Москвы