**ПРОФЕСИОНАЛНА ГИМНАЗИЯ ПО МЕХАНОТЕХНИКА, ЕЛЕКТРОТЕХНИКА, ТЕЛЕКОМУНИКАЦИИ **

**И ТРАНСПОРТ – ШУМЕН**

**ДИПЛОМНА РАБОТА**

**НА**

**Селим Сеферов 12a КЛАС**

**СПЕЦИАЛНОСТ : ПРИЛОЖЕН ПРОГРАМИСТ**

**ТЕМА:**

**Разработка на информационна система „Домоуправител“**

**Изговил: ................... Ръководител: ...................**

**/Селим Сеферов 12А/ /инж. Васил Бояджиев/**

1. Етажна собственост
2. Какво е етажна собственост?

* Етажна собственост има при сгради, които се състоят от самостоятелни обекти -  
  самостоятелни етажи, апартаменти, ателиета, магазини, работилници и гаражи, собственост  
  на различни лица.
* От какво се състои етажната собственост?  
  В сграда с режим на етажна собственост има отделно притежавани помещения, от  
  една страна, и общи части - от друга.  
  Например: Иван Петров е собственик на апартамент No 2 и мазе No 2 в бл. 118, като в  
  нотариалния акт за собственост изрично е отразено, че същият притежава и 5,4 % идеални  
  части от общите части на сградата, в която се намира неговият апартамент.
* Кои са общите части на сградата?  
  Общи части в сградата са тези части, които не обслужват само отделно жилище,  
  магазин или работилница. Общите части не са самостоятелни – те са елементи на сградата.  
  Общите части не могат да бъдат притежание на други, освен на тези, чиято собственост са  
  отделните обекти. Общи на всички собственици са земята, върху която е построена сградата,  
  дворът, основите, външните стени, вътрешните разделителни стени между отделните части,  
  вътрешните носещи стени, колоните, трегерите, плочите, гредоредите, стълбите,  
  площадките, покривите, стените между таванските и избените помещения на отделните  
  собственици, комините, външните входни врати на сградата и вратите към общи тавански и  
  избени помещения, главните линии на всички видове инсталации и централните им уредби,  
  асансьорите, водосточните тръби, жилището на портиера и всичко друго, което по естеството  
  си или по предназначение служи за общо ползуване. Собствениците на отделни обекти в  
  сградата същевременно са съсобственици в общите части на сградата
* Какви са видовете общи части?  
  Общите части са два вида:  
  • Общи части по естеството си:  
  Това са такива елементи на сградата, без които тя не може да съществува като сграда  
  и които с оглед на своя характер обслужват отделните обекти.  
  Това са основите, външните стени, покривът, главните водопроводни и  
  канализационни инсталации, асансьорът и др. такива. Тези общи части не могат да променят  
  предназначението си и нямат самостоятелно значение. Те освен общи не могат да бъдат други.  
  • Общи части по предназначение:  
  Това са онези общи части, без които сградата може да съществува, но все пак те са  
  създадени да обслужват всички собственици и обитатели.  
  Това са дворът, портиерското жилище, сушилнята, пералнята, помещението за детски  
  колички и др. Характерното за тези общи части е, че тяхното предназначение може да се  
  промени. Те могат да преминат в самостоятелна собственост на някой от собствениците или  
  на трети лица.

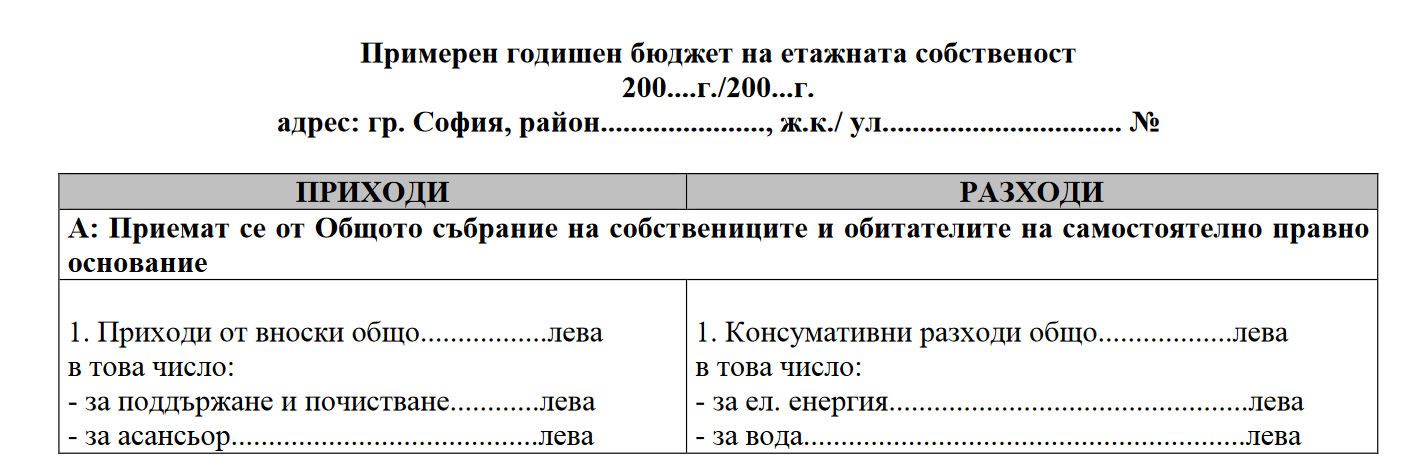
1. Какъв трябва да бъде един домоуправител

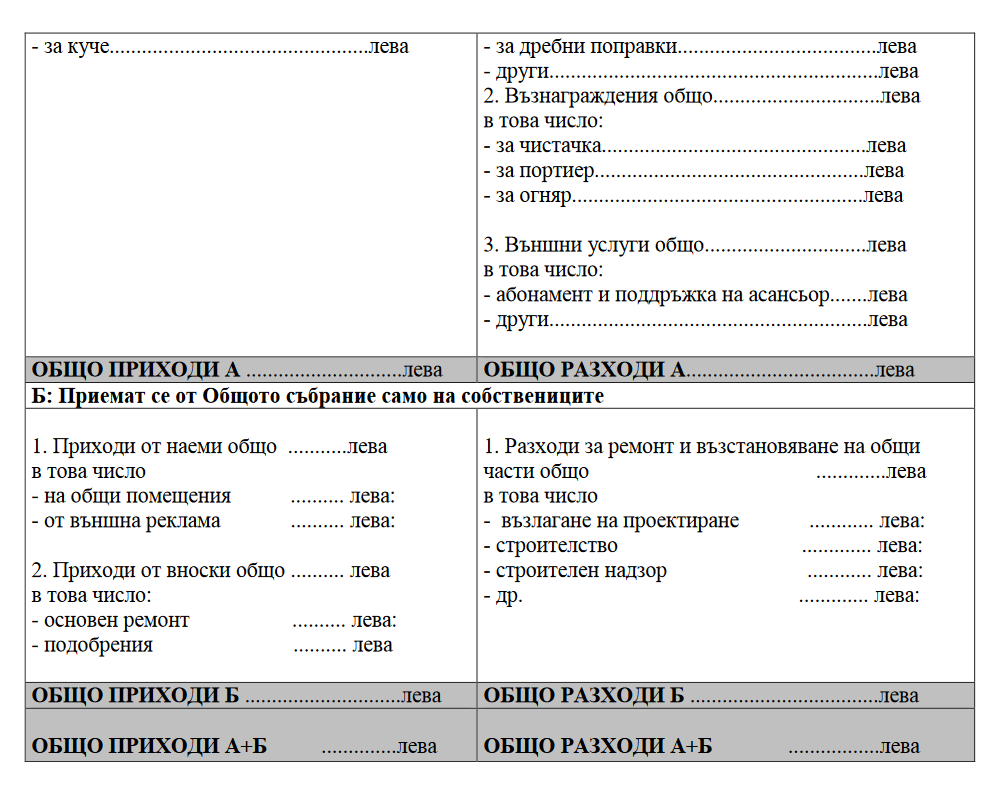
* Какъв е добрият домоуправител?  
  Със сигурност той е един от нас. Човек, който има нашето доверие и уважение. Готов  
  да посвети свободното си време на общите ни проблеми.
* Домоуправителят знае:  
  кои са собствениците в етажната собственост;  
  какво е състоянието на общите части;  
  кои са наемателите в сградата;  
  как да разпредели различните вноски и задълженията между обитателите;  
  колко има в общата каса;  
  какви са изискванията за сигурност и безопасност в сградата;  
  към кого да се обърне, ако възникне проблем.
* Домоуправителят умее:  
  да разговаря с всеки от нас;  
  да балансира между различните интереси в етажната собственост;  
  да ни представлява и защитава пред администрацията и операторите на обществени услуги.
* Домоуправителят прави животът ни заедно:  
  по-подреден;  
  по-малко конфликтен;  
  по-защитен.

1. Как да разпределим общите харчове?

* До края на януари всяка година Общото събрание приема годишен бюджет за приходите и  
  разходите за управлението и поддържането на местата за общо ползване и за общите нужди.  
  Добре е бюджетът да бъде обвързан с отчета за приходите и разходите за изминалата година  
  и да предвижда бъдещите нужди на етажната собственост. За да бъде реалистичен бюджетът,  
  следва да е съобразен с взетите от Общото събрание решения за размера на месечните  
  парични вноски.

1. Бюджетът е задължителен за изпълнение от всички собственици и  
   наематели/ползватели.

* Бюджетът винаги се състои от две части - приходна и разходна. В приходната част се  
  посочват източниците на средства - парични вноски от собствениците и наемателите и  
  приходите от отдадени под наем помещения, пространства или части от тях, които са общи  
  части на сградата. В разходната част на бюджета се предвиждат по пера средствата, които ще  
  се разходват през годината. Това са консумативните разходи, свързани с ползването и  
  поддържането на общите части, и разходите за ремонти и възстановяване на общите части.
* Практиката показва, че особено важно е при определяне на консумативните разходи освен за  
  ел. енергия и вода да се предвидят средства за дребни поправки. Така винаги ще има  
  възможност за подмяна и поправка на ел. крушки, фасунги, електрически контакти, брави,  
  кранове и др

1. Как винаги да знаем колко има в общата каса?  
   Управителят трябва да знае по всяко време какъв е балансът и колко средства има в  
   общата каса. Доброто водене на сметките и правилното разходване на средствата са гаранция  
   за добрите съседски отношения. Както се казва “Чисти сметки – добри приятели”.  
   За да се знае във всеки един момент колко имаме в касата, е задължително воденето на  
   отчетност - какво постъпва и какво излиза от общата каса. За целта избраният от нас касиер  
   трябва да води стриктно книга за разходите и приходите. Всеки извършен разход е  
   необходимо да бъде придружен от съответния разходо-оправдателен документ (фактура,  
   квитанция, разписка, договор и др.). За всеки постъпил приход касиерът издава приходен  
   касов ордер, надлежно оформен, доказващ извършването на съответната вноска.  
   При извършването на дребни ремонти от съкооператори или външно лице, както  
   често става в практиката, е задължително работата да се приеме с протокол, по възможност  
   от независима комисия, избрана от Общото събрание.  
   Например при смяна на улуците на покрива в протокола за приемане се отразява  
   количеството и стойността на разходваните материали, както и качеството на извършената  
   работа.  
   Управителят или управителният съвет са длъжни да отчетат управлението си пред Общото  
   събрание, свикано за избор на нов управител или управителен съвет. Надзорният съвет, а ако  
   няма такъв - определени от Общото събрание лица имат право да извършват проверка на  
   общата каса, като за резултата уведомяват Общото събрание.
2. Създаването, съхраняването и поддържането на книга на етажната собственост

* ОБЩИ ПОЛОЖЕНИЯ

1. Книгата на етажната собственост е документ който съдържа информация за отделните обекти в сградата, собствениците, ползвателите и обитателите. Съставя се от управителя/членовете на управителния съвет/ на етажната собственост, като всяко лице, което прави вписване, освен подписа саморъчно изписва и трите си имена. Книгата може да е с неподвижни страници (във вид на тетрадка) или да е в папка с перфорация (при пренасяне), към която се добавят нови страници. Книгата на етажната собственост се поддържа и съхранява от управителния съвет/управителя на етажната собственост за всяка сграда или вход в режим на етажна собственост.

2. За всеки самостоятелен обект в сградата се съставя отделна партида по образеца, като обектите се подреждат последователно по възходящ ред на номерацията на входовете и обектите. След изписване на тези страници се отбелязва на коя страница на книгата продължава записа за съответния обект и се отделят нови страници, като върху първата от тях се отбелязва, че страницата е продължение за съответен обект. Поради това следва да се осигурят резервен брой комплекти страници за допълнително вписване.

3. Данните по отделните полета се попълват и подържат в актуален вид от управителя на етажната собственост/председателя на управителния съвет или от друго, определено от общото събрание лице.

4. В раздел II, колона 6 от книгата на етажната собственост се отразяват правата и задълженията относно управлението на общите части, в случай че са били уговoрени и отразени в писмен акт между собственика и ползвателя.

5. По смисъла на ЗУЕС ,,домакинство” са две или повече лица, които живеят заедно в един самостоятелен обект или в част от него, в който са регистрирани по постоянен или настоящ адрес, и имат общ бюджет , без оглед на родствената връзка помежду им.

6. По смисъла на ЗУЕС ,,ползватели” са физически или юридически лица, които притежават вещно право на ползване върху чужда вещ, самостоятелен обект в сграда в режим на етажна собственост.

7. По смисъла на ЗУЕС „обитатели” са физически или юридически лица, които не са собственици или ползватели на самостоятелни обекти в сграда в режим на етажна собственост, но пребивават в тях на друго правно основание

* ЗАКЛЮЧИТЕЛНИ РАЗПОРЕДБИ

1. Образецът на книгата е типов. В него задължително фигурират само обособените полета и данните в тях.

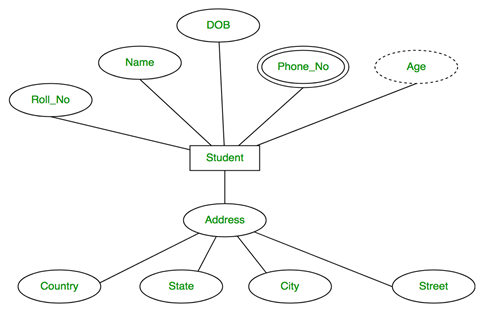
2. Образецът и указанията за създаването, съхраняването и поддържането на книгата на етажната собственост се издават на основание чл. 7, ал. 7 ЗУЕС.

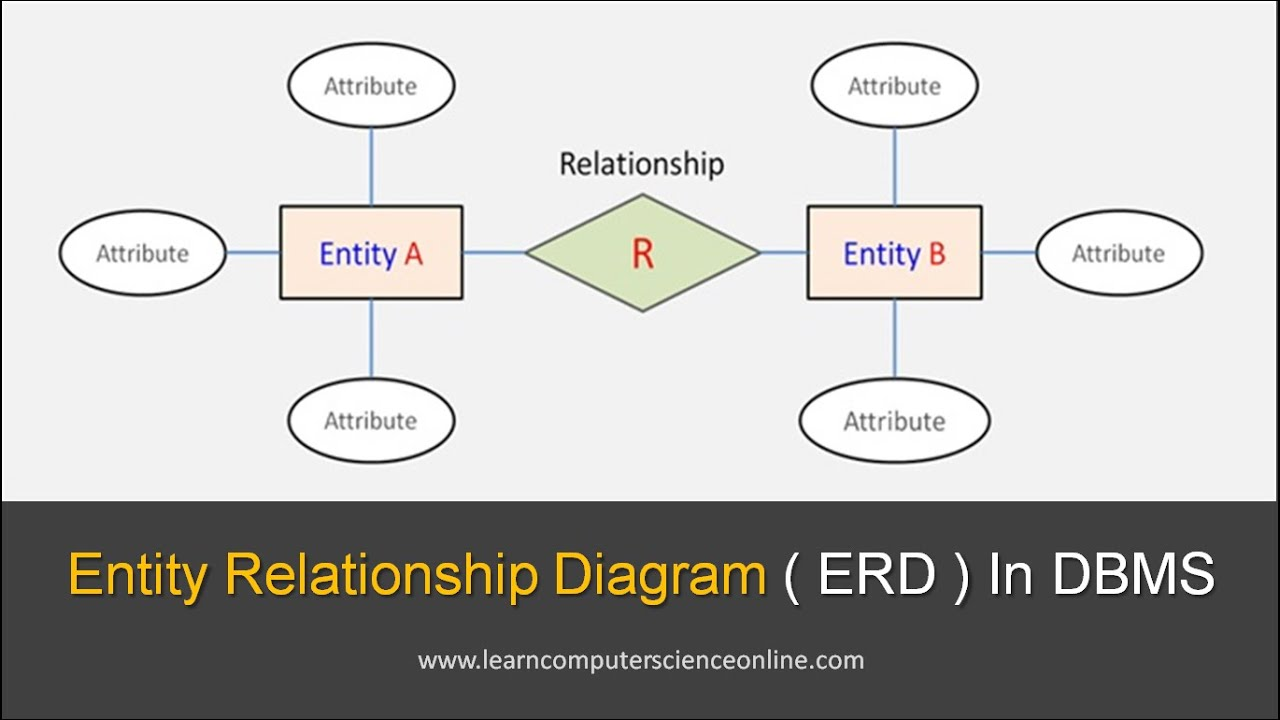
3. В сграда или вход в режим на етажна собственост, в които до влизането в сила на изменението на ЗУЕС с ДВ. бр. 57 от 26 юли 2011 г., се е водила книга на собствениците, може да не се създава нова книга на етажната собственост.

2. Управление на етажната собственост

* Органи на етажната собственост са:
  + общо събрание[[1]](#footnote-1);
  + 2. управителен съвет (УС)/управител.
  + 3. контролен съвет (КС)/контрольор[[2]](#footnote-2).
* Общото събрание на етажната собственост може да приема решения по всички въпроси за вътрешния ред, ползването и поддържането на общите части и др., включително (но не изчерпателно):
  + 1. за приемане, изменение и допълване на правилника за вътрешния ред[[3]](#footnote-3);
* 2. за избор и освобождаване на членове на управителния съвет (управител), касиер, членове на КС(контрольора)[[4]](#footnote-4);
* 3. за приемане на годишен бюджет за приходите и разходите и одобряване на годишните отчети на УС(управителя), както и на КС(контрольора);
* 4. за определяне размера на паричните вноски за разходите за управлението и поддържането на общите части на сградата[[5]](#footnote-5), размера на паричните вноски във фонд ,,Ремонт и обновяване“;
* 5. за приемане на план за извършване на ремонти, реконструкции, преустройства и други дейности в общите части на сградата, включително и по изпълнение на предписаните мерки в доклада към техническия паспорт, или на други предписания на компетентните органи, както и одобрява годишния отчет на УС(управителя) за изпълнението му; за приемане на изменения в плана за извършване на ремонти, когато се налагат непредвидени разходи;
* 6. за извършване на разходи, които са необходими или неотложни за поддържането или за възстановяването на общите части, за извършване на полезни разходи, както и за определяне на размера на разходите за изпълнението на указанията в техническия паспорт;
* 7. за отдаване под наем или за предоставяне за безвъзмездно ползване на общи части на сградата при спазване на нормите за пожарна и аварийна безопасност;
* 8. за обновяване на сградата;
* 9. за предприемане на действия за учредяване на право на ползване, строеж, надстрояване, пристрояване или промяна на предназначението на общи части при спазване изискванията на действащото специално законодателство;
* 10. за поставяне на реклами или технически съоръжения върху сградата;
* 11. за изваждане от сградата на собственик, ползвател или обитател по реда на чл. 45 от Закона за собствеността за определен срок, но не по-дълъг от три години;
* 12. за случаите, когато това е предвидено в нормативни актове във връзка с водоснабдяването, електроснабдяването, топлоснабдяването, газоснабдяването, санитарно-хигиенните норми и предоставянето на други услуги;
* 13. за създаване на условия за достъп в етажната собственост на хора с увреждания[[6]](#footnote-6);
* 14. за използване на общите части на сградата и на прилежащата й площ при възникнали спорове, както и за спазване на вътрешния ред и на санитарно-хигиенните норми[[7]](#footnote-7);
* 15. за присъединяване на сградата към топлопреносната и газоснабдителната мрежа и за прекратяване на топлоснабдяването и газоснабдяването в етажната собственост;
* 16. за опрощаване на финансови задължения, както и за отсрочване или разсрочване на изпълнението им;
* 17. за усвояване на средства от фондовете на Европейския съюз и/или от държавния или общинския бюджет, безвъзмездна помощ и субсидии и/или собствени средства или други източници на финансиране;
* 18. за управление на общите части за получаване на кредити за извършване на необходим или неотложен ремонт, кредити за извършване на полезни разходи или безвъзмездна помощ и субсидии;
* 19. за упълномощаване на УС (управителя) да приеме решение за извършване на неотложни ремонти или на разходи, които не търпят отлагане;
* 20. за възлагане на дейности по поддържането на общите части на сградата на юридическо или физическо лице срещу възнаграждение, като определя и конкретните правомощия на УС (управителя), които могат да бъдат възложени за изпълнение на тези лица.
* Общото събрание не може да откаже приемане на решение за извършване на разходи, които са необходими за поддържането или за възстановяването на общите части на сградата.
* Приетите решения на общото събрание обвързват и нови собственици, ползватели и обитатели на самостоятелни обекти в етажната собственост.

1. Избор на тип база за изграждане на информационна система „Домоуправител“
2. Запознаване с моделите данни в базите данни

* За да бъдат удобни за обработване, данните в компютърната памет трябва да бъдат добре структурирани.
* През 1959 г. се формира международна работна група CODASYL (COmmittee on DAta SYstems Languages, Комитет по езиците за работа с данни). Целта на групата е, да работи върху проблемите на по-широкото използване на компютрите в различните сфери на човешката дейност. Първият резултат от работата на групата е специфицирането на езика COBOL (COmercial Business-Oriented Language, Език за обработка на бизнес данни).
* Възможностите на COBOL за структуриране на данни не могат да задоволят нуждите на изграждащите се ИС и затова специалистите продължават да работят за формулиране на универсални теоретични модели за структуриране на данните.
* Релационен (Табличен) модел
* Основа на релационния модел е двумерният масив – данните са организирани в редове и стълбове. Всички елементи са от един и същи тип, което ограничава възможностите за представяне на данните. Данните в една ИС могат да са от различни типове – текстови, числови, дати, изображения и др. В езиците за ООП този недостатък се преодолява до някаква степен с използването на класове от обекти.
* През 1970 г. Едгар Франк Код публикува своят модел, наречен релационен (табличен), според който данните се групират според типа им и за всяка група еднотипни данни се създава таблица. Така една ИС се състои от много на брой таблици, всяка от които е с еднотипни данни.
* Йерархичен модел
* Йерархии се срещат навсякъде в живота – в организацията на файловете в ОС, в холдингите, в университетите и т.н.
* През 1960 г. компанията IBM разработва йерархичния модел на данните. На негова основа компанията имплементира и разпространява чрез произвежданите от нея компютри IBM360, софтуер IMS (Information Management System) за разработване на ИС с йерархична организация на данните.
* С появата на модела на Код се вижда, че йерархичните структури сравнително просто се симулират в релационния модел. Затова йерархичния модел не получава широко разпространение и такива БД са рядкост.
* Мрежов (Графов) модел
* Графовите структури са най-сложната възможност за организация на данните. Имплементацията е още по-трудна, защото всеки обект от графа може да е свързан с всеки от останалите.
* С все по-широкото и разностранно използване на компютрите се оказва, че има области за които структурирането на данните е в графов модел.
* В днешно време се разработват системи за управление на графови бази от данни с най-различни приложения.
* ER (Entity-Relationship) модел
* Съвременен модел за работа с бази данни е ER-модела (Entity-Relationship model, отношения между обекти). Моделът е създаден от Питър Чен през 1976г. и е разработен за да се избегнат някои недостатъци на класическите модели.
* Спецификациите в този модел се представят в графичен вид с ER-диаграми.
* Данните са моделирани в таблици, наречени ядро на базата;
* Таблиците се състоят от редове/кортежи и колони/атрибути;
* Между таблиците има връзки;



1. Видове бази данни

* Релационни
* Релационните бази данни са един от най-разпространените видове бази данни и се използват от десетилетия. Те използват таблична структура с редове и колони за съхраняване и управление на данни. Данните са организирани в таблици, където всяка таблица представлява тип обект или обект, а редовете представляват отделни екземпляри на този обект. Релационните бази данни използват SQL (Structured Query Language) за заявки и манипулиране на данни.
* Нерелационни (NoSQL)
* NoSQL базите данни не използват традиционна таблична структура и могат да обработват неструктурирани или полуструктурирани данни. Те са проектирани да обработват големи обеми данни и могат да бъдат по-гъвкави и мащабируеми от релационните бази данни. NoSQL базите данни използват различни модели на данни, като базирани на документи, ключ-стойност, семейство колони и базирани на графики.
* Обектно-релационен
  + Обектно-релационен бази данни съхраняват данни като обекти, които могат да съдържат както данни, така и методи. Те са проектирани да работят с обектно-ориентирани езици за програмиране, като Java или C++. Обектно-ориентираните бази данни позволяват сложни връзки между данните и могат да бъдат по-ефективни за определени типове приложения.
* Йерархични
  + Йерархичните бази данни организират данните в дървовидна структура, като всеки запис има един родителски запис и един или повече дъщерни записи. Те често се използват в наследени системи и не са толкова гъвкави, колкото други видове бази данни.
* Мрежови
  + Мрежовите бази данни позволяват по-сложни взаимоотношения между данните чрез използване на подобна на графика структура с възли и ръбове. Те са подобни на йерархичните бази данни, но позволяват множество родителски и дъщерни записи. Мрежовите бази данни бяха популярни през 60-те и 70-те години на миналия век, но до голяма степен бяха заменени от други видове бази данни.
* Графични
  + Графичните бази данни използват графови структури за представяне и съхраняване на данни, като възлите и ръбовете представляват обекти и връзки между тях. Те са особено полезни за данни, които имат много сложни връзки, като социални мрежи или двигатели за препоръки. Графичните бази данни могат да бъдат по-ефективни от други типове бази данни за определени типове заявки.
* Документно ориентирани
  + Базите данни, ориентирани към документи, съхраняват и извличат данни като документи, обикновено под формата на JSON или XML файлове. Те често се използват за уеб приложения, системи за управление на съдържание и мобилни приложения. Базите данни, ориентирани към документи, могат да бъдат по-гъвкави от релационните бази данни и могат да обработват полуструктурирани данни.
* Time-series бази данни
  + Time-series са бази данни, които се променят с времето, като например данни от сензори или данни от финансовите пазари. Те са оптимизирани за съхранение и извличане на големи обеми данни в хронологичен ред. Базите данни с времеви редове стават все по-важни, тъй като Интернет на нещата (IoT) генерира все повече и повече данни с времеви редове.
* Пространствени
  + Пространствените бази данни съхраняват и управляват пространствени данни, като карти или геопространствени данни. Те се използват в различни приложения, като ГИС (географски информационни системи), базирани на местоположение услуги и мониторинг на околната среда. Пространствените бази данни използват специализирани алгоритми и структури от данни за ефективно съхраняване и заявки за пространствени данни.
* In-memory бази данни
  + Базите данни в паметта съхраняват данни в RAM вместо на твърд диск, което позволява по-бързо време за достъп. Те често се използват в приложения, които изискват бързи скорости на четене и запис, като анализи в реално време или системи с голям обем транзакции. Базите данни в паметта могат да бъдат по-скъпи от други видове бази данни поради цената на RAM.

1. Нива на абстракция
   * Абстрахиране – подход за проектиране, реализиране и управление на сложни системи.
   * Специалистите, участващи в процеса се разделят на категории, според профила си. Специалистите от всяка категория определят как ще изглежда системата от тяхна гледна точка и според техните компетенции. Останалите елементи се игнорират, т.е. специалистите се абстрахират от тях.
   * Сложният проект се разбива на нива на абстракция, с всяко от които да се заеме екип от специалисти, чиято компетентност съответства на нивото на абстракция

* Ниво на крайния потребител
  + Най – високото ниво на абстракция.
  + Възложителят се нуждае от софтуер, който да задоволи неговите нужди от информация и да подпомага работата му.
  + Възложителят не познава инструментариума, с който да се разработи ИС, не познава моделите на структуриране на данни, не познава и СУБД, с която да се реализира системата. Той описва нуждите си от информация на естествен език.
  + Възможно е обаче описанието на естествен език да не е достатъчно, за да стартира проектирането на системата.
* Ниво на бизнес анализатора
  + Специалистите по бизнес анализ:
  + Имат способността да вникнат в същността на работата на крайния потребител, да анализират процесите и да определят по-точно нуждите от информация.
  + Владеят инструменти за формализирано описание на информационните процеси и могат да преведат описанието от естествен език в език на инструмента за формализирано описание на ИС.
  + ЕR – моделирането е един такъв инструмент за формализирано представяне на информационните процеси и нужди на крайния потребител.
* Ниво на разработчика
  + Специалисти, които познават инструментите за реализация на базата данни и интегрирането й в ИС. Те превеждат ER – модела на езика на СУРБД, като съпоставят на всеки обект от ER – модела таблица с атрибутите му.
  + Планират и таблиците за осъществяване на връзки между обекти.
  + Добавят атрибути, които бизнес анализатора не е предвидил, защото се е абстрахирал от тях.
  + Изготвят Инструкция за крайния потребител, в която посочват как той да въвежда данни и как да постъпи за получаване на нужните справки.
  + Разработчиците също се абстрахират от някои елементи на системата, които са задължение на системните администратори.
* Аномалии
  + Ако една БД не е нормализирана възникват аномалии, които се проявяват като осиротели записи, сочещи към несъществуващи данни или като невъзможност за добавяне на запис.
  + АНОМАЛИЯ НА ВМЪКВАНЕТО;
  + АНОМАЛИЯ НА ОБНОВЯВАНЕТО;
  + АНОМАЛИЯ НА ИЗТРИВАНЕТО.

1. Избираме релационни бази данни (SQL)
   * Базите от данни се управляват от специален софтуер, който ще наричаме  
     СУБД (система за управление на бази от данни). СУБД представлява мощен  
     инструментариум за създаване и ефективно управление на големи обеми от  
     данни. Данните трябва да се поддържат и съхраняват за толкова време за колкото  
     е необходимо. Освен това, те трябва да се предпазват от неправилен достъп, който  
     може да наруши целостта им (integrity), както и от неправомерен достъп  
     (security).  
     Примери:  
     MS SQL Server, Oracle Database, IBM DB2, MySQL, PostgreSQL
   * Ще въведем и още две нови понятия, свързани със структурата на базата от  
     данни:  
     1. Data Definition Language (DDL) се използва за създаване на схема  
     (описание) на данните. Чрез него абстрактно се дефинира структурата на  
     информацията в базата от данни и се дефинират релации между различните  
     компоненти. Основна грижа на DDL също е да дефинира ограниченията за  
     цялостност на информацията (с цел постигане на интегритет) и да се проектира  
     сигурността на базата от данни.  
     2. Data Manipulation Language (DML) са езиковите средства за обработка на  
     данните. Най-често те се разделят на вътрешни (езика използван от сървъра за  
     базата от данни, например SQL) и външни (езиците за програмиране, например  
     C/C++, Java, и т.н.). Както подсказва самото име, чрез тези езици се подават  
     команди, чрез които сървъра обработва данните.
   * Езикът SQL най-общо казано се използва са създаване на бази от данни,  
     създаване на таблици и връзките между тях, вмъкване на информация, извличане  
     на информация, промяна на информация и изтриване на информация от таблици  
     в база от данни. Въпреки съществуването на официален стандарт на езика, почти  
     всяка система за управление на бази от данни има своя собствена имплементация.  
     Почти винаги се спазват основните правила в езика, но често се добавя  
     допълнителна функционалност, която е специфична за използваната СУБД.  
     SQL се базира на последователност от заявки (команди). Когато част от тези  
     заявки са зависими една от друга говорим за понятието ”транзакция”. Вече  
     споменахме за DDL и DML. Ето как най-общо са въведени тези понятия чрез езика  
     SQL:
   * 1. Data Definition Language:  
     - CREATE DATABASE – създава база от данни  
     - ALTER DATABASE – променя дефиницията на база от данни  
     - DROP DATABASE – изтрива база от данни  
     - CREATE TABLE – създава таблица в база от данни  
     - ALTER TABLE – променя дефиницията на таблица в база от данни  
     - DROP TABLE – изтрива таблица от база от данни  
     - CREATE INDEX – създава индекс  
     - DROP INDEX – изтрива индекс  
     2. Data Manipulation Language:  
     - INSERT – вмъква информация в база от данни  
     - SELECT – извлича информация от база от данни  
     - UPDATE – променя информация в база от данни  
     - DELETE – изтрива информация от база от данни
   * Въпреки, че SQL би трябвало да е унифициран език, всяка система за  
     управление на бази от данни се различава от другите. Много рядко се спазва  
     истинския ANSI стандарт на SQL и често има съществени различия.  
     Както споменахме по-горе, релационните бази от данни съхраняват  
     информацията в таблици. Таблиците се състоят от редове, като всеки ред е един  
     запис в базата, а клетките от таблицата са отделните полета в него. Таблиците  
     имат по една или повече колони, като всяка колона има име и тип. Колоните  
     определят съответните полета в записите от базата. Всяка колона определя типа на  
     поле от запис или типа данни, с които можем да запълним конкретните полета от тази  
     колона.
2. Реализация на проекта

1. Информационна система „Домоуправител“ (ИСД) е специализиран софтуер, предназначен и обслужващ изцяло дейността на професионалните домоуправители - юридически лица. Чрез ИСД специализираните дружества, предоставящи услуги по управление на сгради в режим на етажна собственост, както и фирмите, занимаващи се с управление на имоти, могат цялостно и напълно да обхванат процеса на работа: от една страна с - техните клиенти, и от друга – с доставчиците на услуги.

**РЕШЕНИЕТО ЗА ВАШИЯ БИЗНЕС**

Ако вие сте професионален домоуправител, чрез ИСД можете с лекота да водите и разраствате вашия бизнес, без да се притеснявате от непрекъснато нарастващия обем от данни, които се налага да обработвате.

С помощта на ИСД можете:

* да въвеждате неограничен брой етажни собствености и неограничен брой апартаменти към тях;
* да генерирате с един клик месечните задължения на всеки от апартаментите, съобразно определените бюджетни вноски;
* да изпращате автоматично задълженията на абонатите към ePay/Easypay, както и
* да изпращате уведомление на всеки един от клиентите на електронната му поща;
* да правите справки, отчети и да извличате всякакъв вид обратната информация, с цел пълно следене на: наличности в каси, банкови сметки, задължения на обекти и т.н..

Онлайн интеграцията в реално време с ePay/Easypay, осигурява на Вашите абонати възможност да следят и плащат своите задължения по електронен път или на касите на Easypay. Интеграцията е направена чрез онлайн протокола на ePay, което позволява без човешка намеса, генерираните задължения от PM-Pro да се отразят автоматично в ePay/Easypay, а платените такива да се отчитат незабавно в системата.

Мобилното приложение на ИСД улеснява максимално работата на мобилните касиери, позволявайки им при събиране на вноските на място да следят задълженията на обектите в реално време на всяко мобилно устройство, както и да принтират документ за приетите пари на момента, чрез безжична връзка с мобилен принтер.

Чрез онлайн интеграцията с клиентския портал „ИСД-Клиент“, вие гарантирате на вашите абонати пълна прозрачност, осигурявайки им възможността да следят направените плащания, задълженията си и разходите на етажната собственост.

Заедно с продукта ИСД, ние ви предлагаме и цялостно решение за вашата дейност, което включва:

* перфектно работещ софтуер;
* индивидуални решения, съобразени с вашите специфични нужди;
* обучение на вашите служители;
* непрекъсната онлайн подкрепа по всякакви възникнали въпроси на откритата ни телефонна линия;
* отделно сървърно пространство и гарантирана защита на данните;
* надеждна и бърза техническа поддръжка.

**ОПИСАНИЕ И ВЪЗМОЖНОСТИ**

PM-Pro се състои от четири главни модула:

**1.Административен модул**

Модулът е предназначен за операторите, които ще работят с програмата и основните негови възможности са:

* въвеждане на неограничен брой етажни собствености, които управлявате;
* въвеждане на неограничен брой апартаменти и други обекти към всяка от етажните собствености, с детайлно описание на техните характеристики като: номер, квадратура, брой и имена на живущите и т.н.;
* изготвяне и поддържане на домова книга;
* лесно и бързо изготвяне на месечен бюджет на етажната собственост, с неограничен брой бюджетни пера и възможност за прецизно конфигуриране на разходите за всеки апартамент, в зависимост от това в кои разходи участва и с каква тежест;
* автоматично генериране на бюджетните вноски за всеки апартамент и за всяко бюджетно перо, които при настъпила по-късно промяна могат да бъдат преизчислени с един клик;
* генериране на отделни задължения за всеки тип разход: Фонт ремонт и обновление и Месечна вноска;
* коригиране на вече генерирано задължение или закръглянето му ( до 1, 0.5 или 0.1 ) при приключване на бюджета;
* онлайн интеграция в реално време с ePay/EasyPay, осигуряваща видимост на задълженията в ePay/EasyPay в момента, в който бъде приключен месечния бюджет и автоматично отразяване на плащанията чрез ePay/EasyPay в системата;
* автоматично изпращане на имейл на апартаментите за новогенерираното им задължение;
* пренасяне на бюджет от един месец за друг с един клик, като системата автоматично отчита евентуално направените промени по апартаментите в обекта;
* въвеждане на неограничени по вид и брой разходи и приходи, свързани с генерираните в системата бюджетни пера на етажните собствености;
* водене на отделна сметка за всеки апартамент, която се увеличава автоматично при генериране на задължение и се намалява автоматично при постъпило плащане или платен депозит;
* водене на различни видове сметки и подсметки към тях за всяка етажна собственост, които се намаляват автоматично при извършен разход и се увеличават автоматично при постъпило плащане или платен депозит;
* добавяне и управление на кампании за набиране на средства по конкретни проекти на етажната собственост, вкл. детайлно следене и отчитане на постъпленията и направените разходи по кампанията;
* разпределяне на допълнителните приходи между собствениците по зададени критерии;
* автоматично погасяване на новогенерираните задължения от сметката на апартамента, в случай на налични средства;
* пълно или частично погасяване на всяко едно задължение по всяко време;
* въвеждане на предварително депозирани средства по сметка на даден апартамент за бъдещи задължения, с осигурена възможност за избор дали от нововъведения депозит да се погасят текущите задължения на апартамента или не;
* генериране на различни видове отчети по изпълнението на месечния бюджет с възможност за принт и предаване на етажната собственост;
* генериране на отчети по оператори, апартаменти, отчетни месеци, типове разходи и т.н.

**2. Публичен модул:** ИСД-Клиент

Публичният модул ИСД-Клиентосигурява възможност на абонатите да виждат чрез свободен достъп през интернет своите задължения, плащания, разходите на етажната собственост, както и всякаква друга информация по преценка на професионалния домоуправител.

Порталът е интегриран със системата PM-Pro и в реално време може да извлече информацията за всеки абонатен номер и домоуправител.

Ползването на портала е гаранция за прозрачност и увереност в начина, по който се образуват месечните бюджети на етажните собствености, отговаряйки на въпроса кой колко пари дължи и точно за какво. Това улеснява както клиентите така и професионалните домоуправители в тяхната работа и комуникацията им с живущите.

За ползването на PM-Client абонатът трябва предварително да получи информация за достъп, състояща се от: номер на професионалния домоуправител, абонатен номер и парола.

**3. Контролен модул**

Този модул е предназначен за мениджърите, управляващи дружествата – професионални домоуправители. Чрез него се извършва конфигурирането и достъпа на различните видове потребители в административния модул и мобилното приложение на системата и се задават правата им за работа в тях.

2. Реализация

- Използва се Windows Forms

Windows Forms е UI рамка за изграждане на настолни приложения за Windows. Той предоставя един от най-продуктивните начини за създаване на настолни приложения, базирани на визуалния дизайнер, предоставен във Visual Studio. Функционалност като поставяне на визуални контроли с плъзгане и пускане улеснява създаването на настолни приложения.

С Windows Forms вие разработвате графично богати приложения, които са лесни за внедряване, актуализиране и работа, докато сте офлайн или докато сте свързани с интернет. Приложенията на Windows Forms имат достъп до локалния хардуер и файловата система на компютъра, на който се изпълнява приложението.

- За база се използва MySQL

MySQL е популярна система за управление на релационни бази данни (RDBMS) с отворен код, която използва Structured Query Language (SQL) за управление на данни. Той се използва широко за уеб приложения и поддържа различни платформи, включително Windows, Linux и macOS. MySQL предоставя функции като мащабируемост, висока достъпност и сигурност и може да обработва големи количества данни ефективно. Освен това е съвместим с много езици за програмиране и може да се интегрира с различни софтуери и инструменти. Като цяло MySQL е многофункционален и надежден избор за управление на данни в различни приложения.

1. Литература

* <https://www.syncfusion.com/forums/145624/winforms-controls-in-net-framework-4-7-and-4-8>
* <https://learn.microsoft.com/en-us/dotnet/desktop/winforms/overview/?view=netdesktop-7.0>
* ВАНЯ ЯНЕВА
* <https://developercommunity.visualstudio.com/t/cant-use-mdf-file-or-logldf-that-was-created-on-an/1547664>
* <https://docs.oracle.com/en-us/iaas/mysql-database/doc/getting-started-mysql-database-service.html#GUID-3C1DA346-CDA2-442E-9D22-A7C244B9DF04>
* <https://dobritesasedi.bg/resources/domoupravitel-narachnik.pdf>
* <https://delarbg.com/d48/wp-content/uploads/2018/04/NUES-Web.pdf>

1. [↑](#footnote-ref-1)
2. [↑](#footnote-ref-2)
3. [↑](#footnote-ref-3)
4. [↑](#footnote-ref-4)
5. [↑](#footnote-ref-5)
6. [↑](#footnote-ref-6)
7. [↑](#footnote-ref-7)