Конструирование программного обеспечения

Горшков Ян К-14.1

**Диаграмма вариантов использования**

Диаграмма вариантов использования показывает функциональные возможности программной системы. Диаграмма изображена на рисунке 1.

Основные прецеденты:

* Неавторизований користувач – любой пользователь зашедший на сайт, не прошедший авторизацию;
* SQL Server – БД для хранения информации;
* Пацієнт – пользователь, прошедший стандартную процедуру регистрации
* Лікар – пользователь-врач
* Лаборант – пользователь-лаборант
* Модератор лікарні – пользователь-модератор
* Адміністратор клініки – пользователь-администратор клиники
* Адміністратор системи – главный администратор, который добавляет, удаляет клиники, лаборатории, аптеки, и назначает администраторов клиник.
* Тех. підтримка – пользователь, отвечающий за техническую поддержку
* Новини – новости всей системы в целом
* Лікарні – список всех больниц, которые сотрудничают с системой
* Новини лікарні – новости конкретной больницы
* Прайс-лист – цены на услуги в больнице
* Мед. персонал – список врачей, работающие в данной больнице и график работы.
* Історія – история больницы
* Інша інформація – остальная информация, связанная с больницей
* Аптеки – все аптеки, которые сотрудничают с системой
* Налаштування – настройки пользователя
* Змінити пароль – изменение пароля пользователя
* Змінити e-mail – изменение почтового ящика пользователя
* Змінити зображення – изменение фотографии профиля пользователя
* Реєстрація – процесс регестрирования пациента в системе.
* Авторизація – процесс входа пользователя в систему
* Записатися на прийом – осуществить запись на приём к врачу
* Вибір лікаря – выбор врача, к которому будет осуществлена запись
* Перегляд вільного часу – вывод на экран свободного времени для записи
* Обрати дату – выбор даты записи
* Обрати час – выбор времени записи в данный день
* Вибір симптомів – указание симптомов
* Перегляд особистої мед.картки – возможность посмотреть результаты приёмов врачей и обследования в лабораториях
* Друк – печать выбранных страниц мед.карты
* Зберегти у PDF – сохранение мед.карты в pdf-формате
* Перегляд мед. картки – просмотр медицинской карты пациента врачем или лабораторией
* Висновок лікаря – после обслуживания пациента, врач записывает в мед.карту диагноз, рекомендации, препараты и т.д.
* Перегляд черги та симптомів пацієнтів – просмотр всех записавшихся пациентов, их симптомов, и предполагаемый системой диагноз
* Висновок лабораторії – описание результатов обследования и загрузка архива с графической информацией.
* Реєстрація модератора – регистрация модератора больницы
* Видалення модератора – удаление модератора больницы
* Реєстрація мед. працівника – регистрация мед. работника в больнице
* Видалення мед. працівника – удаление мед. работника в больнице
* Додавання лікарні – добавление больницы в систему
* Видалення лікарні - удаление больницы из системы
* Призначення адміністратора лікарні – регистрация администратора выбранной больницы
* Додавання лабораторії – добавление лаборатории в систему
* Видалення лабораторії - удаление лаборатории из системы
* Додавання аптеки - добавление аптеки в систему
* Видалення аптеки - добавление лаборатории из системы
* Додати новину – добавление новости
* Видалити новину – удаление новости
* Видалити коментарії – удаление комментария пользователя
* Редагувати прайс-лист – изменение прайс-листа
* Редагувати інформацію – изменение информации об учреждении
* Звернутися у тех. підтримку – обращение пользователя в службу поддержки
* Відповісти – ответ работника тех. поддержки на письмо пользователя



Рисунок 1 – Диаграмма вариантов использования

**Диаграммы последовательности действий**

Процесс регистрации начинается со входа в соответствующее меню сайта – Регистрация. После чего, пользователь заполняет все поля. После подтверждения введенных данных происходит POST-запрос на веб-сервер, на котором происходит проверка на валидность данных. При успешном результате проверки, веб-сервер посылает запрос к базе данных, который добавляет новые данные, после чего будет осуществлен автоматический вход в систему.

Процесс авторизации также начинается с ввода пользователя логина и пароля. После проверки веб-сервера на валидность будет осуществлен запрос в БД, суть которого- нахождение пользователя по логину и паролю. Если была найдена соответствующая запись, то запрос возвращает пользователя, и будет осуществлен вход в систему. Если записи не был найдено, то на экран будет выведено сообщение о неверном вводе логина или пароля. Диаграмма изображена на рисунке 2.



Рисунок 2 – Диаграмма последовательности действия – регистрация и авторизация

Следующая диаграмма показывает часть действий, которые может совершать администратор системы. Такими действиями являются: добавление и удаление больницы, добавление и удаление лаборатории. Действия добавления похожи друг на друга, и ограничиваются вводом текстовой информации, после чего происходит запрос на добавление в БД. При удалении учреждения, администратору выводится на экран список, в котором он выбирает нужное. После выбора, происходит запрос на удаление записи из БД.

Диаграмма изображена на рисунке 3.



Рисунок 3 - Диаграмма последовательности действия – администратор

**Кооперативные диаграммы**

Кооперативная диаграмма для процесса регистрации и авторизации изображена на рисунке 4. Нагрузка на элемент = 4

****

Рисунок 4 - Кооперативная диаграмма – регистрация и авторизация

Кооперативная диаграмма для действий администратора изображена на рисунке 5. Нагрузка на элемент = 5



Рисунок 5 – Кооперативная диаграмма – администратор

**Диаграммы классов**

Первая диаграмма классов описывает модель для хранения данных, используя ORM ADO.NET Entity Framework. Каждый класс описывает таблицу в БД. Поля классов соответствуют атрибутам таблицы. Связи классов – связи между таблицами. Агрегация указывает связь один к одному, и один ко многим. Диаграмма изображена на рисунке 6. На данной диаграмме присутствуют такие классы:

* DataBaseContext – главный класс-контекст, который отвечает за доступ ко всем классам сущностям в виде строго типизированных классов-коллекций. В конструктор передаётся имя тега файла конфигурации XML, где прописана строка подключения к БД.
* DataBaseInitializer – класс, необходимый для инициализации БД (подход Code First).
* Institution – класс, от которого наследуются все учреждения (аптеки, больницы, лаборатории)
* Pharmacy – класс, отвечающий за хранение информации об аптеке
* Clinic – класс, отвечающий за хранение информации о больнице
* Laboratories – класс, отвечающий за хранение информации о лаборатории
* Role – класс, хранящий информацию и роли пользователя
* User – класс, хранящий информацию о пользователе
* Doctor – класс, хранящий доп. информацию о враче
* WorkTime – класс, хранящий график работы врача
* BusyTime – класс, содержащий информацию о записи пациента
* DoctorReport – класс, отвечающий за хранение отчета врача
* LabaratoryReport – класс, отвечающий за хранение отчета лаборатории
* News - базовый класс для общих признаков новости
* MainNews - класс, отвечающий за хранение общей новости
* ClinicNews – класс, отвечающий за хранение новости клиники
* Comment – класс, отвечающий за хранение комментария



Рисунок 6 – Диаграмма классов - Model

Вторая диаграмма классов описывает контроллеры системы. Диаграмма изображена на рисунке 7. На данной диаграмме присутствуют такие классы:

* MainController – контроллер, отвечающий за общую функциональность, доступную всем типам пользователей.
* CabinetController – контроллер, отвечающий за управление личным кабинетом пользователя
* ModeratorController – контроллер, отвечающий за функциональность модератора
* DoctorController – контроллер, отвечающий за функциональность доктора
* ClinicAdminController – контроллер, отвечающий за функциональность администратора клиники
* AdminController – контроллер, отвечающий за функциональность администратора системы
* PatientController – контроллер, отвечающий за функциональность пациента
* SupportController – контроллер, отвечающий за функциональность тех. поддержки



Рисунок 7 – Диаграмма классов – контроллеры

**Диаграммы состояния**

Состояние регистрации начинается с ввода пользователем информации. После ввода происходит чтение данных из формы (ФИО, Дату рождения, фотографию, почту, логин и пароль). После этого происходит проверка всех введенных данных на валидность – соответствует ли строка фамилии, имени, или отчества длине от 3 до 30 символов. Выбрана ли дата рождения, введен ли корректныц e-mail адрес, соответствует ли логин и пароль длине от 6 до 30 символов. При несоответствии условию будет выведено соответствующее сообщение и ожидание повторного ввода. Если данные введены корректно, то происходит запрос в БД. При возникновении ошибки выводится соответствующее сообщение. Если проблем не возникло, то состояние будет завершено. Данная диаграмма изображена на рисунке 8.



Рисунок 8 – Диаграмма состояний – регистрация

Состояние авторизации начинается с ввода логина и пароля пользователем. После ввода происходит проверка данных на их наличие. При отсутствии данных будет выведено соответствющее сообщение и ожидание повторного ввода. При корректном вводе будет проивзеден запрос в БД. Если запрос не вернет запись, то будет выведено сообщение о том, что логин или пароль введены неправильно. При возврате запросом записи будет начата сессия, и произойдет переадрессация пользователя в его личный кабинет. Данная диаграмма изображена на рисунке 9.



Рисунок 9 - Диаграмма состояний – авторизация

**Диаграмма компонентов**

Данная диаграмма показывает разбиение классов на различные пространства имён (namespaces). Все классы-модели, отвечающие за хранение данных, находятся в пространстве имён DigitalMedicine.Model. В данном пространстве имен есть ещё несколько: Model.Users, Model.News. Все контроллеры проекта находятся в пространстве имен DigitalMedicine.Controllers. Все представления находятся в пространстве имён DigitalMedicine.Views. Данное пространство имён состоит из множества других, соответствующие контроллерам. Данная диаграмма изображена на рисунке 10.

****

Рисунок 10 – Диаграмма компонентов

**Диаграмма размещения**

Данная диаграмма показывает способ развертывания программной системы и его работу в реальных условиях. В данной системе есть центральный веб-сервер (IIS), который связан с сервером баз данных по локальной сети. Для доступа клиентам к веб-приложению происходит обращение к веб-серверу по глобальной сети. Веб-сервер формирует HTML-страницу. Данная диаграмма изображена на рисунке 11.



Рисунок 11 – Диаграмма размещения