МИНИCTEPCTBO НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

**Кафедра прикладной информатики**

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 13-14

ДИСЦИПЛИНЫ «Программная инженерия»

НА ТЕМУ:

Реинжиниринг информационных систем

**Выполнил:**

студент группы ПИН-б-о-21-1

ФИО Джабраилов Тимур Султанович

Проверил: ассистент кафедры прикладной информатики

Щёголев А.А.

Ставрополь, 2024

**Задание 1**

Проведём анализ информационной системы «Ekampus» для Северо-Кавказского Федерального Университета.

|  |  |
| --- | --- |
| Разработчик | Северо-Кавказский Федеральный Университет |
| Требования к системе | * Простой и понятный дизайн * Обладание функциональными возможностями для абитуриентов, студентов, преподавателей, а также внутренних сотрудников университета * 99% доступность * Надежность * Безопасность данных пользователей |
| Стоимость | 15 млн руб. |
| Возможности | * Возможность абитуриентов подавать заявления на поступление * Возможность студентов просматривать расписание, предметы, оценки. * Возможность студентов прикреплять выполненные работы, а также переписываться с преподавателем в чате. * Возможность преподавателей просматривать расписание, предметы, оценки, списки студентов. * Возможность преподавателей проставлять посещаемость студентов. |
| Примеры внедрения | «Ekampus» внедрён и активно используется в «Северо-Кавказском Федеральном Университете» |
| Недостатки | * Отсутствие официальной мобильной версии * Недостаток агрегированной статистической информации студента * Отсутствие выдаваемых студентам работ. (Для этого приходится использоваться сторонними файлообменниками) |

**Задание 2**

Составим список возможностей пользователей в информационной системе «ekampus»

|  |  |
| --- | --- |
| Тип пользователя | Возможности |
| Абитуриент | * Просматривать список факультетов и требований * Подавать заявление на поступление |
| Студент | * Просматривать расписание студента * Просматривать расписание преподавателя * Просматривать предметы * Просматривать оценки * Прикреплять выполненные работы |
| Преподаватель | * Просматривать расписание преподавателя * Просматривать преподаваемые предметы * Проставлять оценки * Просматривать выполненные работы * Проставлять посещаемость |
| Внутренний сотрудник университета | * Настройка расписания |

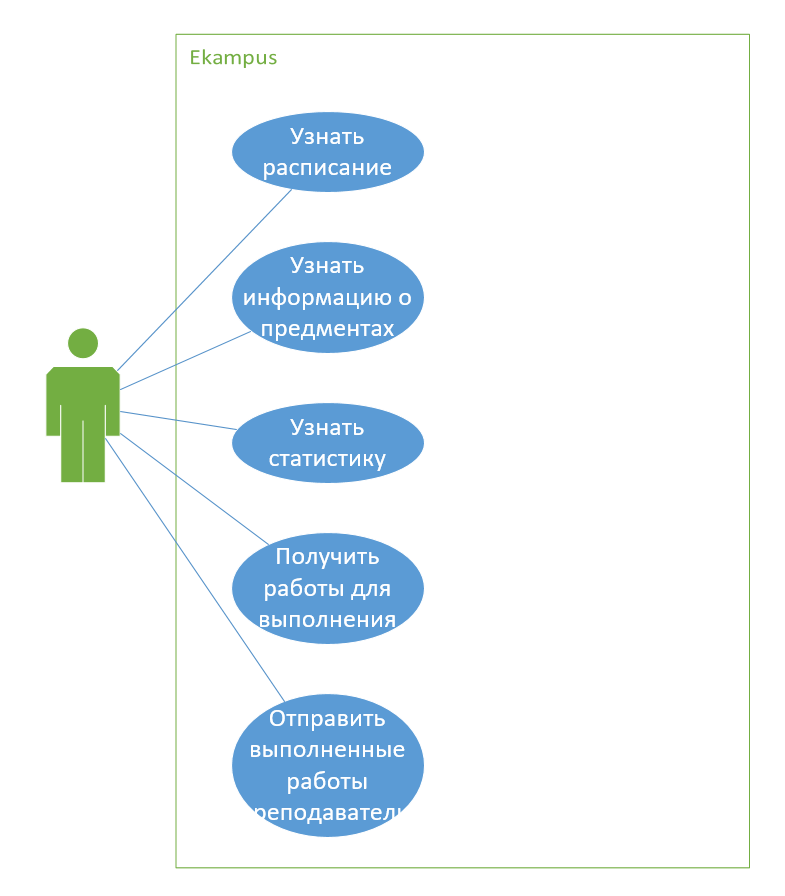
**Задание 3**

Составим список требований студента:

* Узнавать расписание (студент, преподаватель)
* Узнавать информацию о предметах (студента, преподавателя)
* Узнавать статистику (оценки, пропуски, рейтинг)
* Получение выполняемых работ
* Отправление выполненных работ преподавателю

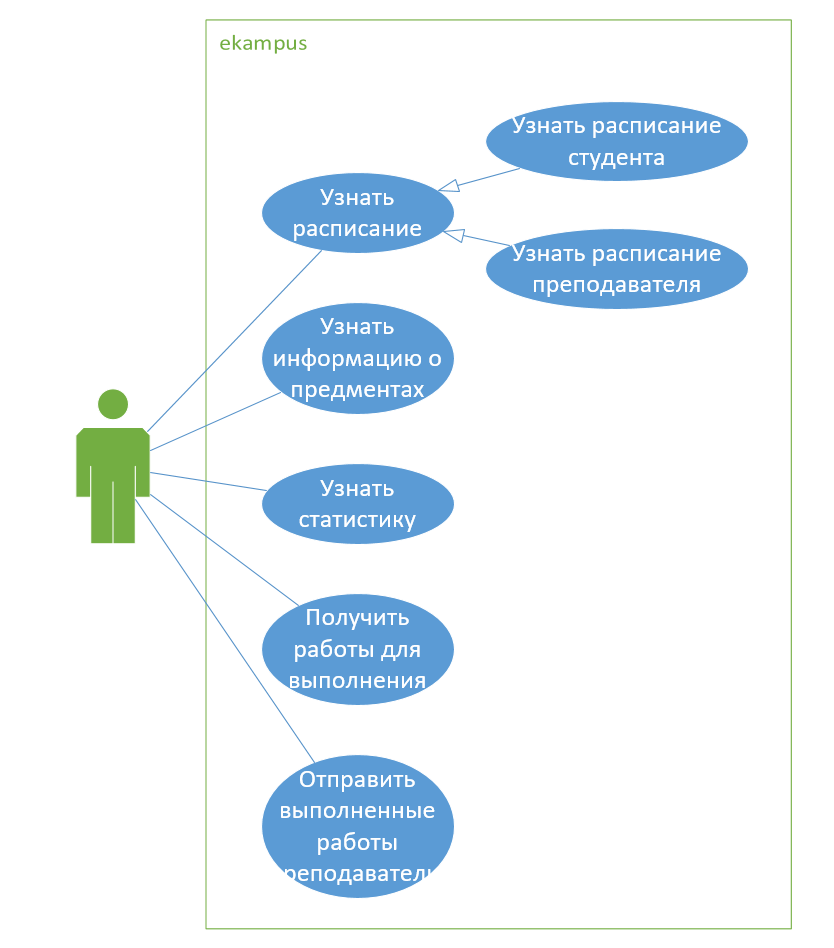
**Задание 4**

Теперь составим диаграмму прецедентов на основании сформированных требований



**Задание 5**

Детализируем прецедент «Узнать расписание». Данный прецедент является обобщением прецедентов узнать расписание студента и узнать расписание преподавателя.



**Документация для информационной системы университета**

**Прецедент 1: Просмотр расписания студента**

**Описание:**

Данный прецедент позволяет студенту просматривать свое расписание занятий на текущий семестр.

**Действия:**

1. **Авторизация:**
   * Студент вводит свой логин и пароль на странице авторизации.
   * Система проверяет корректность введенных данных.
   * Если данные верны, система перенаправляет студента на главную страницу.
2. **Просмотр расписания:**
   * На странице расписания отображается информация о:
     + Названии дисциплины;
     + Преподаватель;
     + Типе занятия (лекция, лабораторная работа);
     + Номер пары;
     + Аудитории.
   * Студент может фильтровать расписание по неделям

**Прецедент 2: Просмотр расписания преподавателя**

**Описание:**

Данный прецедент позволяет преподавателю просматривать свое расписание занятий на текущий семестр.

**Действия:**

1. **Авторизация:**
   * Преподаватель вводит свой логин и пароль на странице авторизации.
   * Система проверяет корректность введенных данных.
   * Если данные верны, система перенаправляет студента на главную страницу.
2. **Просмотр расписания преподавателя:**
   * На странице расписания преподавателя отображается информация о:
     + Названии дисциплины;
     + Группе;
     + Типе занятия (лекция, лабораторная работа);
     + Номер пары
     + Аудитории.
   * Студент может фильтровать расписание по неделям.

**Дополнительные сведения**

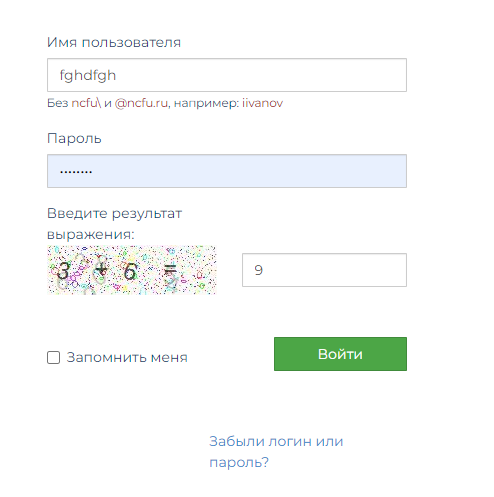
* Система должна быть доступна 24/7.
* Система должна быть защищена от несанкционированного доступа.
* Система должна быть отказоустойчивой.
* Система должна соответствовать требованиям безопасности.

**Технические характеристики**

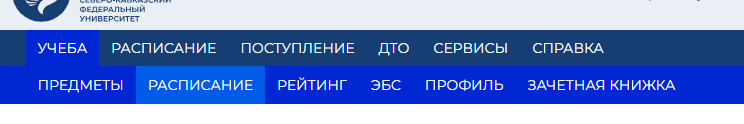
* Платформа: ASP.NET
* Язык программирования: C#
* База данных: SQL Server
* Веб-сервер: nginx

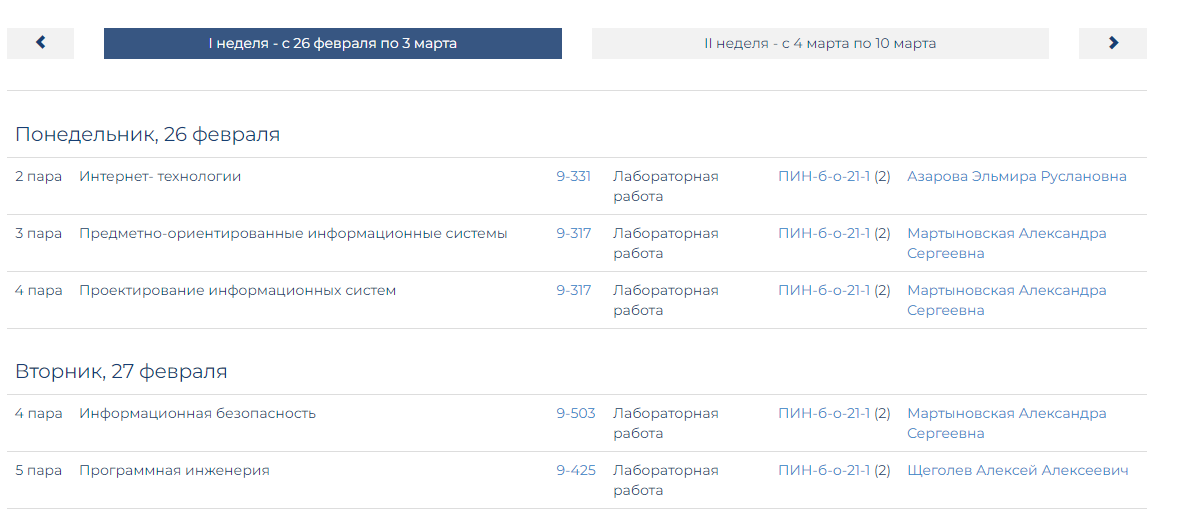
**Задание 6**

Составим сценарий работы с использованием print screen-ов продукта. Для начала необходимо студенту авторизоваться.



Затем, чтобы посмотреть расписание необходимо перейти на страницу расписание в разделе учёба.





Таким образом, студент может просматривать своё расписание на текущую неделю или же на следующую.

**Задание 7**

У студентов может возникнуть потребность в агрегированной статистике о его пропусках, о его успеваемости, просадки в знаниях и автоматических рекомендациях по улучшению знаний в той или иной области, также в современном мире возникает необходимость мобильной версии информационной системе и отсутствует возможность преподавателям прикреплять задания для студентов, из-за чего требуется пользоваться сторонними файлообменниками, следовательно, были выявлены небольшие недостатки системы т.к. она не предоставляет такой информации для студентов.

**Вывод**

В ходе данной лабораторной работы был осуществлён реинжиниринг информационной системы «ekampus», были обозначены возможности и требования, построены диаграммы прецедентов, а также были выявлены недостатки информационной системы.