|  |
| --- |
| **ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - СОФИЯ**  **Факултет – „Компютърни системи и управление“** |

**КУРСОВ ПРОЕКТ**

**Дисциплина – „Програмни среди“**

***Студент: Иван Николаев Колев, фак. № 121213114, гр. 43***

***Специалност: „Компютърно и софтуерно инженерство“***

**Дата: 31.05.2016 г. Заверил:**

**/ гл. ас. д-р А. Ташева /**

**София  
2016**

Съдържание

[1. Въведение 3](#_Toc452449873)

[1. Възникване на идеята: 3](#_Toc452449874)

[2. Описание на проблема 3](#_Toc452449875)

[3. Описание на решението: 3](#_Toc452449876)

[4. Описание на предимствата: 3](#_Toc452449877)

[2. Анализ на съществуващи разработки 4](#_Toc452449878)

[3. Проектиране 4](#_Toc452449879)

[4. Реализация 4](#_Toc452449880)

[5. Потребителско ръководство 4](#_Toc452449881)

[6. Заключение 4](#_Toc452449882)

[7. Литература 4](#_Toc452449883)

[8. Приложение 4](#_Toc452449884)

# 1. Въведение

Текущият проблем който разглеждаме е свързан със запазването на зали за провеждане на учебни занятия (лекции, упражнения, лабораторни) и изготвянето на дигитален график удобен за ползване от студенти и преподаватели.

## 1. Възникване на идеята:

Идеята е плод на ситуации, при които преподавателите не се уведомени, че графика на залата в която трябва да водят лекции или упражнения е променен и там се обучава друга група, или когато преподавателя лично е сменил залата, но не всички студенти са уведомени – тогава настава хаос, неразбирателство и закъснения.

## 2. Описание на проблема

Сегашният процес по промяна на разписанието на залите, в случая в който преподавател променя залата - разчита на предаване на информацията от уста на уста (преподавател информира един или няколко студенти, да разпространят на другите си колеги, че залата и часът се променя) или в най-добрия случай – изпращане на email до някоя от групите засегнати от промяната. А в случая, когато друг преподавател изиска зала и промени графика – първият, който е трябвало да ползва залата, трябва да отиде до кабинета на охраната, да прегледа хартиените списъци с предадените ключове и да прецени коя зала е свободна в момента, за да я заеме, но това отключва възможността за възпрепиятстване на занятията на друг преподавател по веригата, понеже няма как всеки да знае часовете за групи на всички останали и това е много сериозен организационен проблем.

## 3. Описание на решението:

Решението което предлагам е интегриране на централизирана уеб услуга, която да бъде консумирана от мобилни, настолни или уеб приложения (Mobile, Desktop, Web clients). Идеята е всеки да може да достъпва графика през смартфона си, през уеб приложение или през настолен компютър, за да следи най-актуалните промени по графиците на всяка от залите, а преподавателите (които ще бъдат администратори в тази система) да ползват лесен интерфейс за промяна или добавяне на нови събития към графиците на залите.

## 4. Описание на предимствата:

По този начин, за по-малко от 20 секунди един преподавател може да актуализира графика на залата за конкретен ден, като успоредно с това има достъп до вече запазените часове, като всичката информация е събрана на едно място и се избягва нуждата от търсене във всичките 200 pdf файла, които в момента се ползват за седмичен разпис на часовете на отделните групи от всички факултети. А студентите, могат лесно да проверяват актуализирания график директно от мобилните си устройства или от персоналните си компютри. Така, вината за това че някой не е в час с новата програма, пада изцяло върху него.

Тази система позволява по-удобно запазване на часове и извън седмичния разпис, предоставен от факултетната администрация. Например при нужда за отработване на дадено упражнение, могат да се запазят зали за някой от почивните дни, без да се случват застъпвания с други групи инцидентно отработващи занятия. Или просто запазването на зали за събития, които не са в пряка връзка с обучителния процес – външни лектори, вътрешни събития само за академичния състав, събрания, форуми (Career fair).

# 2. Анализ на съществуващи разработки

Обикновено това са три или повече разработки, като се изтъкват положителни и отрицателни страни в тяхната реализация и потребителски опит

# 3. Проектиране

Описание на всички аспекти от крайния продукт

Графики, ER диаграми, Workflow диаграми и др.

Кой ще използва продукта? – Насочва ни към функционалностите, дизайна и данните, които ще използва нашето решение.

Какви данни ще се използват? (Анализ на входните данни от потребителя, изграждането им в структури – класове и обекти, съхранението им (криптиране на пароли) SecureString-ове за sensitive данни, описание на бизнес процесите в приложението.

Описание на потребителския интерфейс, навигация сред страниците.

# 4. Реализация

Тук се описват с текст и примери от кратки фрагменти код практическите реализации, които извършвате по зададения в „Проектирането“ предварителен план. Като всеки от горните аспекти се показва как се реализира: ER диаграмите – в бази данни, класове и обекти; бизнес процесите - програмен код; потребителският интерфейс – използваните контроли(intellibox autocomplete, timepicker custom implementation), реализация на валидацията и др.

# 5. Потребителско ръководство

Тук показвате как вашият потребител трябва да може да взаимодейства с вашия продукт. – Описание на преходи от различни страници, резултати от работата със системата и др.

# 6. Заключение

заключавате дали вашият продукт е готов за употреба, дали среща нуждите на вашите потребители и дали е конкурентноспособен на съществуващите решения

# 7. Литература

номерирани и надлежно описани източници, ползвани при анализа, проектирането и разработката на приложението ви

# 8. Приложение

По-важните части от кода