



Софийски университет "Св. Климент Охридски"
Факултет по математика и информатика

Проект

по

**Софтуерни архитектури и
разработка на софтуер**

на тема

***UniDigit – Информационна
система за управление на
процесите и студентската
информация в университет***

Изготвен от:

Добромир Колев, 62356

Белла Събева, 62188

Съдържание

1. Въведение

1.1. Организация на текущия документ

1.1.1. Предназначение на документа

1.1.2. Описание на структурите

1.1.3. Структура на документа

1.2. Общи сведения за системата

1.3. Архитектурни драйвери

1.3.1. Свързани с функционалността на системата

1.3.2. Свързани с качеството на системата

1.4. Разширен терминологичен речник

1.4.1. Списък на софтуерните елементи

1.4.2. Всички други специфични термини

2. Декомпозиция на модулите

2.1. Общ вид на декомпозицията на модули за системата

2.2. Клиенти

2.2.1. Предназначение

2.2.2. Основни отговорности в системата

2.2.3. Подмодули

2.3. Бизнес логика

- 2.3.1. Предназначение**
 - 2.3.2. Основни отговорности в системата**
 - 2.3.3. Подмодули**
 - 2.4. Модул на администратора**
 - 2.4.1. Предназначение**
 - 2.4.2. Основни отговорности в системата**
 - 2.4.3. Подмодули**
 - 2.5. Външни източници**
 - 2.5.1. Предназначение**
 - 2.5.2. Основни отговорности в системата**
 - 2.6. База данни**
 - 2.6.1. Предназначение**
 - 2.6.2. Основни отговорности в системата**
 - 2.6.3. Подмодули**
- 3. Описание на допълнителните структури**
 - 3.1. Структури на процесите**
 - 3.1.1. Обща структура на процесите**
 - 3.1.1.1. Мотивация за избор**
 - 3.1.1.2. Първично представяне**
 - 3.1.1.3. Описание на елементите и връзките**
 - 3.1.1.4. Описание на обкръжението**

3.1.1.5. Описание на възможните вариации

3.1.2. Структура на процеса

“Регистрация на организатор”

3.1.2.1. Мотивация за избор

3.1.2.2. *Първично представяне*

3.1.2.3. Описание на елементите и връзките

3.1.2.4. Описание на обкръжението

3.2. Структура на внедряването

3.2.1. Мотивация за избор

3.2.2. *Първично представяне*

3.2.3. Описание на елементите и връзките

3.2.4. Описание на обкръжението

3.2.5. Описание на възможните вариации

4. Архитектурна обосновка

1. Въведение

1.1. Организация на текущия документ

1.1.1. Предназначение на документа

Документът цели да представи архитектурата на информационна система за управление на процесите и студентската информация в университет.

1.1.2. Описание на структурите

Архитектурата включва в себе си следния набор от структури:

- **Декомпозиция на модулите**

Структурата представя системата, разделена на отделни смислови и функционални единици, наречени модули. Те са общо пет на брой. *Clients* модулът е този, който обособява потребителите на системата. *Business logic* е модула, който се грижи за бизнес логиката и включва в себе си подмодули, които обособяват отделни функционалности на системата от гледна точка на клиентите и на администратора, като администраторът има функционалности, които са достъпни само за него. Включен е и модул за *обработка на грешки*. *External systems* модулът предоставя възможност на потребителите да се снабдят с достъп до частни и държавни външни системи, които могат да бъдат други специализирани сайтове. *Database* модулът отговаря за съхранението на данните на системата, свързани с потребителите, специалностите, курсовете, оценките на студенти, събитията и комуникацията.

- **Структура на процесите**

Представя нагледно последователността на изпълнение на някои от по-важните процеси в системата.

- **Структура на внедряването**

Изобразява взаимовръзките между хардуерните и софтуерните елементи на системата и по-специално връзката със сървъра и базата от данни, където се съхранява информацията за потребителите и конкурсите.

1.1.3. Структура на документа

- **Въведение**

Дава описание на целта на документа, неговата структура, както и използваните термини и структури при изграждане на софтуерната архитектура на системата.

- **Декомпозиция на модулите**

Представя системата като съвкупност от смислови и функционални единици, наречени модули. Всеки модул е описан детайлно заедно със своите подмодули.

- **Допълнителни структури**

В тази секция се представят още две структури на системата - структура на процесите и структура на внедряването, като за всяка една от тях се описва причината за избор, първичното ѝ представяне и нейните елементи и връзки.

- **Архитектурна обосновка**

Обосновават се причините за взетите решения по време на разработката на архитектурата на база на изискванията и използваните тактики.

1.2. Общи сведения за системата

Системата цели да предостави възможност на университетите да управляват процесите и студентската информация в определен университет. Тя е предназначена за такъв тип потребители, които се интересуват от това да си осигурят най-подходящите стоки и услуги за техните нужди. Функционалностите, заложили в нея, са свързани с това лесно да може да се публикуват курсове и специалности, да се търсят и да се кандидатства по тях чрез студентски профили. Освен това на студентите се дава възможност, в случай че се интересуват, да се снабдят с информация за своите оценки. Потребител има достъп до организиране на събития и комуникация с останалите. За да се подсили сигурността на данните на всеки потребител, се прилага криптиране на данните и преминаване през посредник - мениджър на базата данни, който да гарантира защитата на

информацията. Освен това, като допълнително гарантиране на консистентността на данните се преминава и през защитна стена. Отказоустойчивостта на софтуерния продукт се поддържа чрез допълнителни сървъри, които да се включват в случай на прекалено високо натоварване в системата и да работят паралелно.

1.3. Архитектурни драйвери:

1.3.1. Свързани с функционалността на системата

- **Регистриране на потребителски профил. Обслужване на отдели в системата**

Системата обслужва няколко отдела – учебен, счетоводен, студентски съвет и административен. Всеки отдел предполага определен тип потребители, както и администратор на системата, преподавател и студент. При регистриране на профил студентите и преподавателите записват своите данни и допълнителна информация за своите компетентности. Базата от данни ще се изгради посредством акаунтите на участниците в портала, което е основополагащо за изграждането на целостта на информационната система.

- **Одобряване на учебни планове и записване на курсове**

За да се използва архитектурата на системата по главно предназначение, тя трябва да позволява на преподавателите да предлагат учебни планове и курсове на потребители от учебния отдел, а те от своя страна след приемане да поискат одобрение от административния отдел. Архитектурата ще използва уникален драйвер, който позволява на студент да запише курс, само ако отговаря на специалността му, и ако данните му съответстват на входните изисквания за компетентност.

- **Контрол и поддръжка на финансови операции**

Важно за нормалното функциониране на една университетска система е безопасното реализиране на финансови операции под строг контрол на счетоводния отдел. За целта е включен специален софтуер, който поддържа електронно плащане с външни платформи като раурал, easурау, както и по банков път. Предвидените надеждни начини за извършване на платежност ще се използват както от потребители на системата, така и за осъществяване на трансфери с външни изпълнители на услуги.

- **Поддръжка на електронни студентски книжки**

Необходима функционалност, която архитектурата предоставя, е възможност за притежаване на електронна книжка като част от студентския

профил. Системата разрешава редактиране само от преподавателски профили по записаните от студента курсове. Достъпът за преглед на книжка е ограничен. Притежават го само съответния студенски профил, преподавателските, които водят записаните от потребителя курсове, административен отдел и естествено администраторът на системата.

- ***Комуникация между потребителите***

Важна част от фундаменталното предназначение на архитектурата на системата е да предоставя възможност за организиране на събития (курсове, състезания и групи с обща цел) от всички потребители и API за генерираните събития, както и интегриране на специален драйвер за осъществяване на комуникация между потребителите в архитектурата на системата.

1.3.2. Свързани с качеството на системата

- ***Защита и сигурност на системата***

За да се гарантира легитимността и сигурността на архитектурата на системата е нужно тя да спазва няколко изисквания за качество. Кореспонденция с Държавни публични регистри за изпращане информация (която се контролира от учебен и административен отдел) за статуса на студентите, връзка със система за контрол на НАП и възможност за комуникация с други платформи при усъвършенстване на архитектурата на системата. За защита на потребителите се предвижда проверка на личните данни при вход в портала. Поради наличие на сензитивна информация се предприемат мерки за предпазване от неортозиран достъп чрез включване на утвърдени се модели за сигурност, както и съхранение на копие на данните на друг сървър, за да се предотврати евентуално изгубване.

- ***Поддръжка и устойчивост на системата***

Порталът ще бъде наличен през целия период на своето съществуване, освен при планирано обновяване по време на официални празници. Това изискване ще се удовлетвори чрез осигуряване на високоскоростна връзка с интернет и висока наличност на сървърите. Архитектурата на системата е предвидена надеждно да издържа на пикови натоварвания, каквито се очаква да възникнат, чрез приоритизиране на заявките и внимателното им последователно обработване без видима поява на забавяне.

1.4.Разширен терминологичен речник

1.4.1.Списък на софтуерните елементи

1.4.1.1.Clients

1.4.1.1.1.Lecturer

1.4.1.1.2.Student

1.4.1.1.3.Student Council

1.4.1.1.4.Accounting Department

1.4.1.1.5.Academic Department

1.4.1.1.6.Administration

1.4.1.1.7.Admin

1.4.1.2.Business logic

1.4.1.2.1.Main functionality

1.4.1.2.1.1.Common User Manager

1.4.1.2.1.1.1.Create Events

1.4.1.2.1.1.2.Communication

1.4.1.2.1.2.Students and Lecturers Manager

1.4.1.2.1.2.1.Students

1.4.1.2.1.2.1.1.Create a Profile

1.4.1.2.1.2.1.2.Discipline Enrolment

1.4.1.2.1.2.1.3.Course Enrolment

1.4.1.2.1.2.1.4.Recieve a Grade

1.4.1.2.1.2.2.Lecturers

1.4.1.2.1.2.2.1.Create a Profile

1.4.1.2.1.2.2.2.Grade

1.4.1.2.1.3.Others Manager

1.4.1.2.1.3.1.Financial Operations

1.4.1.2.1.3.2.Approve Disciplines and Courses

1.4.1.2.1.3.2.1.Academic Department

1.4.1.2.1.3.2.2.Administration

1.4.1.2.2.Admin

1.4.1.2.2.1.Data Authentication

1.4.1.2.2.2.Statistics

1.4.1.2.2.3.Publish Study Courses

1.4.1.2.2.4.User Managment

1.4.1.2.3.Error handling

1.4.1.2.4.Notification sender

1.4.1.2.5.Database manager

1.4.1.3.External systems

1.4.1.4.Database

1.4.1.5.1.Communication

1.4.1.5.2.Users

1.4.1.5.3.Specialty Disciplines

1.4.1.5.4.Grades

1.4.1.5.5.Courses

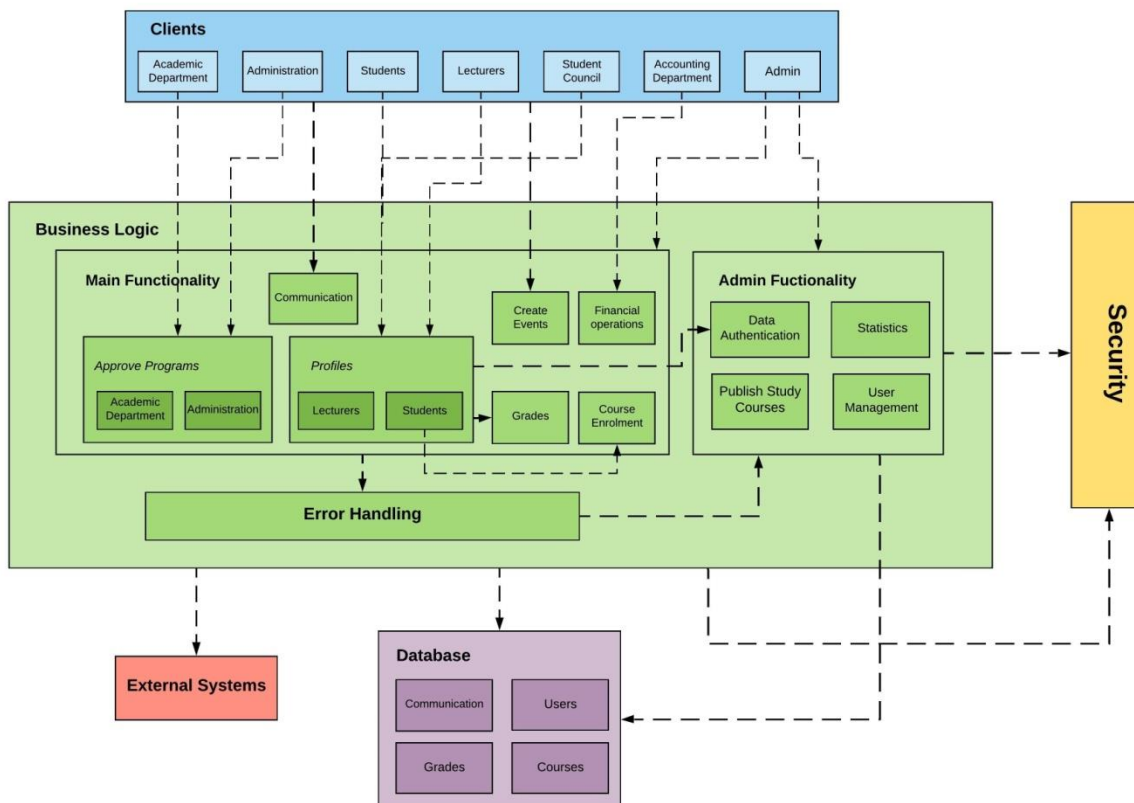
1.4.1.5.6.Events

1.4.2.Всички други специфични термини

- **Софтуер** – набор от машинно-четими инструкции, които управляват действията на компютърния процесор
- **Софтуерна система** – комуникиращи помежду си компоненти, които са част от една компютърна система
- **Външна система** – отдалечена софтуерна система, която е източник на данни или функционалности, които се използват от настоящата система
- **База от данни** – колекция от информация, която е така организирана, че да може лесно да се достъпва, управлява и актуализира
- **Кеш-памет** - спомагателна памет за ускоряване обмена на данни между различните нива в йерархията на паметта.
- **Криптиране** - кодиране на значението на информация
- **Потребител** – човек, който използва компютърна или мрежова услуга
- **Сървър** – стартирана инстанция на софтуерна система, която може да приема заявки от клиент и да връща подходящи отговори
- **Клиент** – част от компютърна или софтуерна система, която достъпва услуга, предоставена от сървър
- **Интерфейс (interface)** – споделена граница, между която два отделни компонента на компютърна система си обменят информация
- **Модул** – логически обособена софтуерна единица
- **Процес** - съвкупност от стъпки, която изгражда логическо действие и стига определена цел
- **Декомпозиция** - софтуерна структура, показваща как системата се разделя на отделни модули. Типовете елементи изграждащи тази структура са модули, а връзките между тях са от типа “X е подмодул на Y”.
- **Структура на внедряване** – архитектурна структура, която показва връзката между софтуера и хардуера

2. Декомпозиция на модулите

2.1.Общ вид на декомпозицията на модули за системата



2.2.Клиенти

2.2.1. Предназначение

Модулът представя основните потребители, които очакваме в системата - клиентите. Те се разделят на няколко групи – > Лектори, Студенти, Студенти от студентски съвет, Счетоводен отдел, Учебен отдел, Администрация и Администратор.

2.2.2. Основни отговорности в системата

Основната отговорност на този модул е да помогне за по-лесното разбиране на работата на системата и да представи нагледно ясно различимата трислойна архитектура на софтуерния

продукт, като за целта модулът се явява представител на ползвателите на системата.

2.2.3.Подмодули

2.2.3.1. Учебен отдел

Потребителите от учебен отдел имат достъп до създаване на събития и комуникация с други потребители. Също така имат право да приемат предложения за академични програми.

2.2.3.2. Администрация

Потребителите от администрацията получават основните функционалности – създаването на събития и комуникация с потребителите в системата. Освен това, те могат да одобряват курсове

2.2.3.3. Студенти

Студентите разполагат с достъп както до организиране на събития и комуникация с други потребители, така и до записване на курсове и преглед на своите книжки (оценки).

2.2.3.4. Професори

Професорите ще имат право на основните функционалности и оценяване на студентите, записали техните предмети.

2.2.3.5. Студентски съвет

Потребителите на студентския съвет разполагат с права над основните функционалности и студентски права.

2.2.3.6. Счетоводен отдел

Потребителите от счетоводния отдел ще имат достъп до основните функционалности и контрол на финансовите операции в системата.

2.2.3.7. Администратор

Администраторът ще разполага с достъп до всички функционалности на системата, но предимно ще отговаря за коректната употреба на основните функционалности и изпълняване административната функционалност в системата.

2.3.Бизнес логика

2.3.1.Предназначение

В модула за бизнес логиката са включени подмодули, които се разграничават в отделни функционалности на системата от гледна точка на клиентите. В него се съдържа и модулът за обработка на грешки.

2.3.2.Основни отговорности в системата

Задачите на този модул са да извърши действията, които са заложиени като цели за работата на сайта по предназначение, да се свързва с базата от данни и с външни източници, за да се снабди с необходимата информация.

2.3.3.Подмодули

2.3.3.1.Основни функционалности

2.3.3.1.1 Одобряване на програми

2.3.3.1.1.1. Учебен отдел

Потребителите от учебен отдел приемат предложения от преподавателите за специалностите и курсовете.

2.3.3.1.1.2. Административен отдел

Потребителите на административния отдел одобряват приетите предложения от учебния отдел за учебни планове и програми.

Описание на интерфейсите:

```
public void Approve(string username1,string username2,string  
program)
```

Одобрение на предложена програма

*Вход:Потребител,който предлага
програмата,потребител,който я приема,програма.*

2.3.3.1.2.Профили

Системата разполага с драйвер за поддръжка профили на студенти и преподаватели, в които се записват техните лични данни, както и информация за техните умения и академични постижения.

2.3.3.1.2.1. Професори

Професорските акаунти разполагат с информация за техните компетентности и функционалност за оценяване на студенти.

2.3.3.1.2.2. Студенти

Студентските профили разполагат с информация за техните компетентности, както и възможност за записване на курсове.

Описание на интерфейсите:

- `public bool Login(string username, string password);`

Вход: Потребителско име, парола

Изход: Булева стойност за успешно вписване

- `public void Register(string username, string password, string type)`

Регистрация на потребител

Вход: Потребителско име, парола, тип потребител

- `public void Edit(string username, string information)`

Редактиране на информацията на даден потребител

Вход: Потребителско име, информация

2.3.3.1.3. Създаване на събития

Системата включва механизъм за публикуване на събития като курсове, състезания и д.р., които могат да се организират от всички потребители.

Описание на интерфейсите:

- `public void createEvent(string username, string event)`

Създаване на събитие от потребител

Вход: потребител, събитие

2.3.3.1.4. Записване на курс

Системата използва драйвер, който позволява на студент да запише курс, само ако отговаря на специалността му, и ако данните му съответстват на входните изисквания за компетентност.

Описание на интерфейсите:

- `public bool courseEnrolment(string username, string course, string requirements)`

Вход: Потребителско име, курс, изисквания за вписване

Изход: Булева стойност дали записването е било успешно

2.3.3.1.5. Оценяване

Преподавателите имат право да нанасят оценки на студентите, записали техните дисциплини, в електронната им студентска книжка,

която е част от студентския профил.

Описание на интерфейсите:

• *public bool Grades(string lecture, string student, string course, double grade)*

Лекторът вписва оценка в електронната студентска книжка на студента по съответния курс

Вход: лектор, студент, курс, оценка

2.3.3.1.6. Финансови операции

Потребителите от счетоводния отдел ще управляват безопасно финансовите операции, които засягат другите потребители с помощта на софтуер, поддържащ електронно плащане с външни платформи като paypal, easypay, както и по банков път.

2.3.3.1.7. Комуникация

Този модул дава възможност за обмен на лични съобщения между всички потребители в системата.

2.3.3.2. Функционалност на администратор

2.3.3.2.1 Мениджмънт на потребители

Администраторите се стремят към правилната употреба на платформата. Ако потребител злоупотреби с полето на действие на правата си, се забранява временно достъпът му до системата.

2.3.3.2.2 Публикуване на курсове за обучение

Този модул е основен за функционалността на системата.

Администраторите ще публикуват курсове за обучение, които ще участват в съставянето на учебни планове за обучение. Курсовете се създават така, че студентите имат право за записване само на курсове, съвместими с тяхната специалност и компетентност.

2.3.3.2.3 Проверка на данни

При регистрация на студентски или лекторски профил се изисква справка на предоставените данни с лични документи. След проверката на администратор, те получават съответните права за ползване на системата, а данните се съхраняват в базата от данни на системата. Системата изисква потребителско име и парола на потребителите при всяко влизане.

2.3.3.2.4 Статистика за трафика на системата

Администраторите системно следят статистика за трафика и потреблението на платформата през различни интервали от време, отчитат натовареността на системата, а също и тенденциите на повишаване и понижаване интереса към нея.

Описание на интерфейсите:

- `public List<User> DataAuthentication(string username,string data)`

Справка на предоставените данни с лични документи

Вход:потребител,данни

- `public List<User> GetUsersStatistics(datetime periodStart, datetime periodEnd,);`

Справка за посетителите на системата за дадения период

Вход: начало на период, край на период

2.3.3.3. Обработка на грешки

В този модул за отказоустойчивост на системата се предвижда обработка на грешки, които могат да настъпят по време на регистрация, оценяване, създаване на събитие и записване на курс. При критично събитие администраторът се уведомява под формата на имейл.

2.4 Сигурност

2.4.1.Предназначение

Този модул съществува, понеже системата трябва да издържа на пикови натоварвания и безотказно да предоставя защита на личните данни на потребителите, предпазвайки от неоторизиран достъп към тях.

2.4.2 Основни отговорности в системата

Основната отговорност на този модул е да отговаря за достъпността, сигурността и отказоустойчивостта на системата.

2.5 Външни системи

2.5.1 Предназначение

Чрез този модул платформата установява връзка с няколко външни системи: Държавни публични регистри, на които се изпраща информация за статута на студентите, системата за контрол на НАП и системата на Мудъл за управление на учебното съдържание. Порталът предоставя възможност за лесни модификации с цел корелация с пробиващите нови технологии.

2.5.2 Основни отговорности в системата

Задачата на този модул е да се свързва с други външни системи за обмен на информация.

2.6 База от данни

2.6.1. Предназначение

В модула за базата от данни се съхраняват нужните на системата данни, свързани с потребителите, курсовете, оценките и комуникацията.

2.6.2. Основни отговорности в системата

Базата от данни има за цел да съхранява личната информация на всеки от регистрираните потребители, както и тази за събитията, които са публикувани на страницата на сайта.

2.6.3. Подмодули

2.6.3.1. Потребители

В тази база се пазят личните данни на потребителите - имена, имейл адрес, телефон, извършени плащания и т.н.

2.6.3.2. Курсове

Съхраняват се данни за всички одобрени и изчакващи одобрение курсове. Тези данни включват: име на академик, описание и изисквания на програмата, брой места за курса.

2.6.3.3.Оценки

Пазят се електронните студентски книжки с всички оценки по дисциплините на даден студент.

2.6.3.4. Комуникация

Базата от данни съхранява историята на проведената комуникация между потребителите, т.е. дава възможност за преглед хронологията на съобщенията.

3.Описание на допълнителните структури

3.1.Структури на процесите

3.1.1. Обща структура на процесите

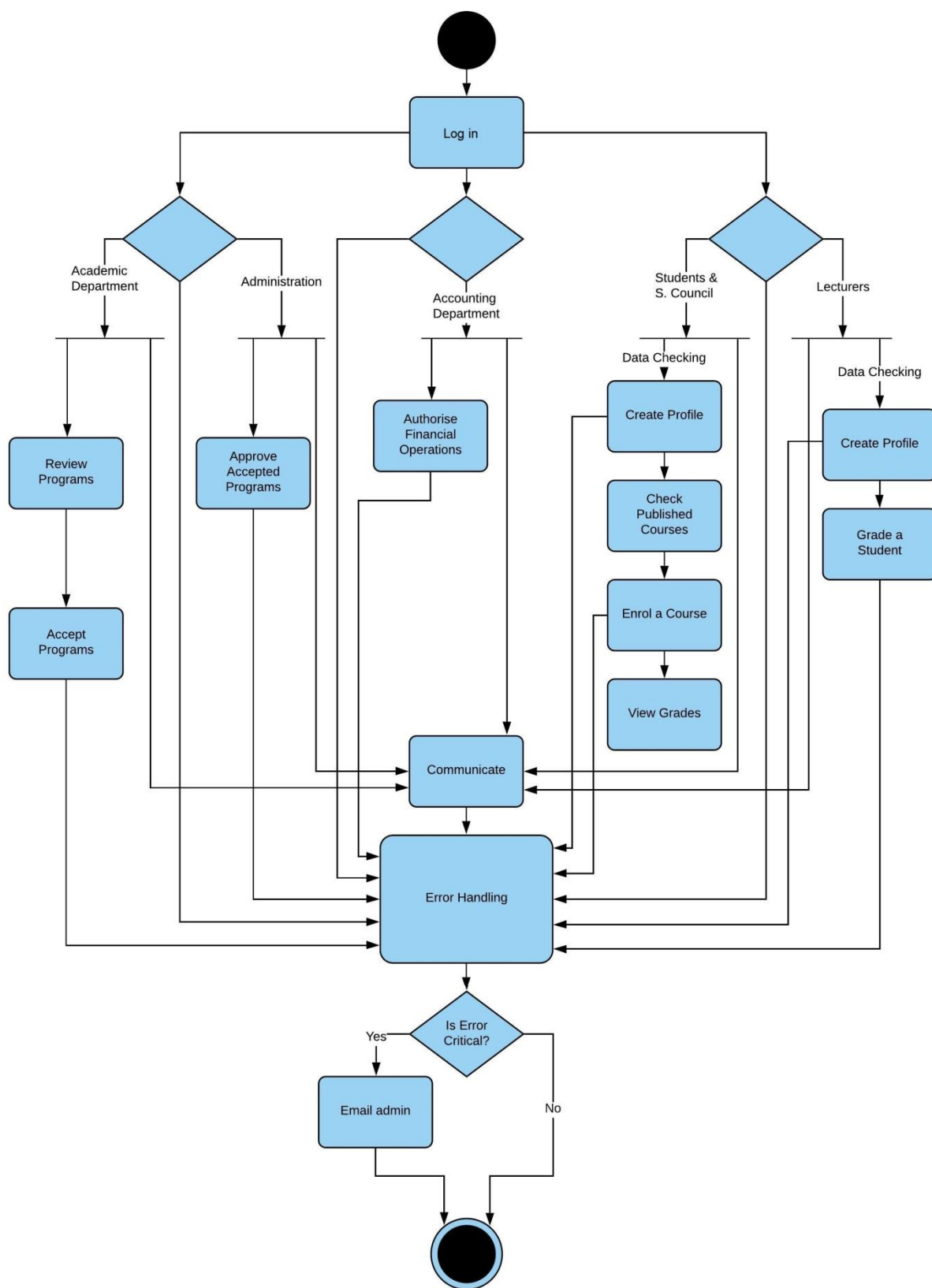
3.1.1.1. Мотивация за избор

Структурата на процесите представя по-динамичен и задълбочен поглед върху процесите, които се реализират в системата. Тя визуализира как компонентите (процесите) се свързват помежду си чрез комуникационни, синхронизационни и блокиращи канали.

Структурата на процесите представя нагледно последователността на изпълнение на процесите и е изключително важна и полезна, тъй като има отношение към бързодействието, високата надеждност и ефективност на системата.

3.1.1.2.Първично представяне





3.1.2.3. Описание на елементите и връзките и

3.1.2.4 Описание на обкръжението

При първоначалното влизане в системата се извършва процес на регистрация и в зависимост от това дали тя е на лице от учебен отдел, администрация, счетоводен отдел или студент, преподавател се предприемат различни действия. В случай, че лице от учебен отдел се впише в системата, то получава право да преглежда предложенията от преподавателите за учебни планове и програми, да ги приеме и предаде към администрацията. Ако лице от администрация се включи в системата, то има възможност да одобрява приетите от учебен отдел специалности и курсове. При вход в системата на лице от счетоводен отдел, то получава достъп до орторизиране на финансови операции в системата. Ако за първи път се регистрират студент или професор първо трябва да създадат собствени профили с проверена информация за себе си. След това студентите имат право да преглеждат курсове и техни анотации, да се записват спрямо компетентностите от собствения профил и да преглеждат електронните си книжки с оценки по записаните дисциплини. След създаване на профил лекторите могат да попълват оценки на техни студенти в електронните им книжки. Всички потребители имат възможност да използват комуникативен канал за лични съобщения между потребителите на системата. При настъпване на определен тип грешка се преминава към процеса за справяне с грешките, който в краен случай уведомява по имейл администратора на системата.

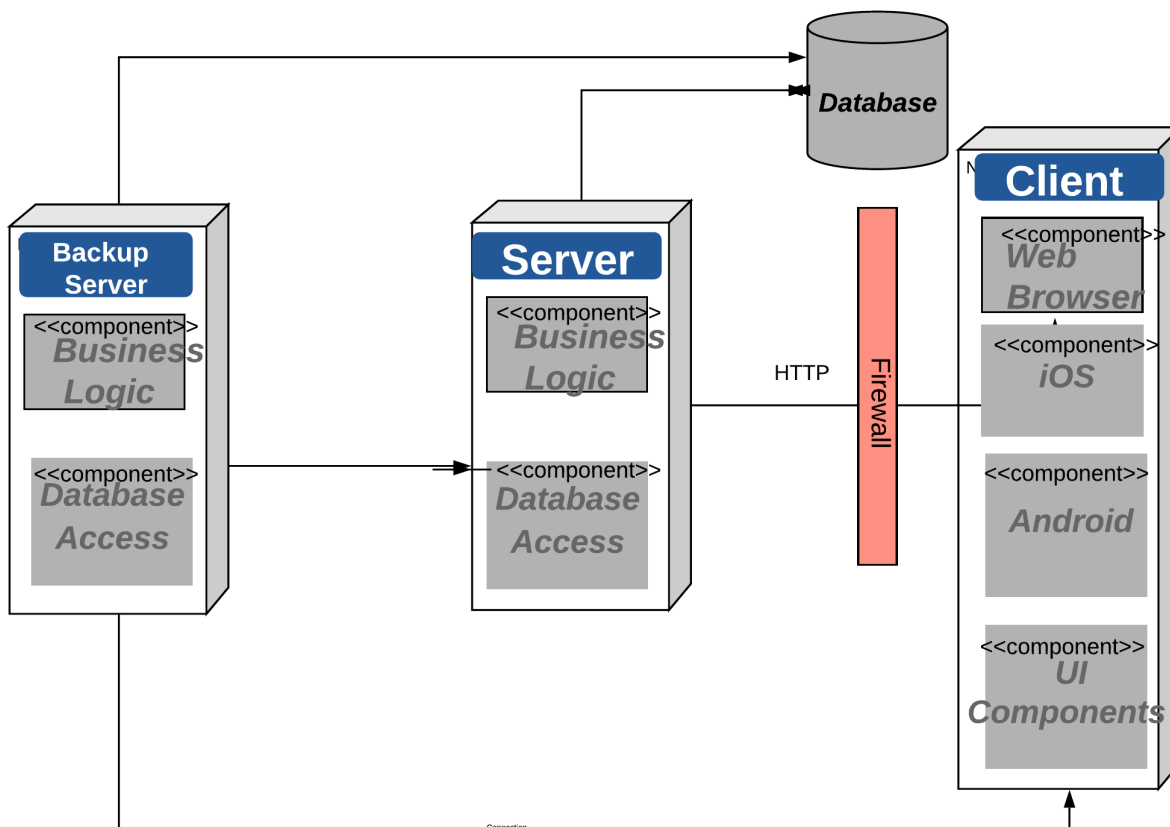
3.2. Структура на внедряването

3.2.1. Мотивация за избор

Структурата на внедряването е посочена, като една от допълнителните структури, за да покаже как софтуерът се разполага на хардуера и какви са комуникационните канали между устройствата. Структурата на внедряването помага за по-добра организация на модулите по време на разработка и прави по-лесен процеса на разпределение на работата по екипи, тъй като освен на модули, системата е ясно разделена и на хардуерно ниво. Изготвянето на тази структура е част и от определянето на какви и колко големи хардуерни ресурси трябва да бъдат на разположение за

реализирането на системата. Тези ресурси са съществени при определянето на себестойността на разработването и поддръжката на системата.

3.2.2.Първично представяне



3.2.3 Описание на елементите и връзките

- **Client Tier** – Предоставя потребителския интерфейс на системата – чрез брауър или на мобилно устройство. Свързва се със сървърната част с помощта на компонентите -Web browser, iOS, Android и Client Tier. Освен това съдържа компонент-сбор от UI Components. В него се намира графичния потребителски интерфейс, който клиента ще използва.
- **Server** – Това е сървър и в него е разположена самата същност на системата, състои се от Business Logic и Database Access.
- **Backup Server** – дубликат на Server, както се вижда и от първичното представяне на структурата (Backup Application Server). Този резервен сървър отговаря за осигуряването на по-добра наличност и производителност при големи натоварености на системата или при срыв на основния сървър. Двата сървъра работят на принципа на пасивен излишък, като главния сървър отговаря за синхронизацията на състоянието на резервния. Затова има и връзка между двата сървъра, чрез която да се осъществи самата синхронизация.

- **Firewall**- С помощта на HTTP протокол се изпращат заявки към сървъра и се получава отговор от него. Преди заявките да достигнат сървъра преминават през Firewall, за да се увеличи сигурността на системата.

- o **Business Logic** – В него са представени всичките функционалности на системата, като например одобряване на програми ,създаване на профили,създаване на събития,оценяване. Това е най-важния компонент от всички. Той осъществява връзката между Server-а и Client Tier-а.

- o **Database Access** – Този компонент си комуникира с Business Logic. Той взима данните и информацията от тях и ги пренасочва към базата от данни, който е разположен на отделния сървър-Database.

3.2.4. Описание на обкръжението

Сървърът може да общува и с външни за системата ресурси (специализирани сайтове), откъдето да разменя допълнителна информация относно потребителите и финансовите операции.

3.2.5. Описание на възможните вариации

Логиката на приложението може да се разполага на един сървър или да бъде дубликирана на няколко сървъра, които да обработват заявки успоредно.

4.Архитектурна обосновка

Архитектурата на системата е проектирана с цел да бъдат изпълнени функционалните и нефункционалните изисквания. Използвана е трислойна архитектура - клиент, сървър, база данни. Всеки модул и съответните му подмодули отговарят за отделна функционалност. Тъй като потребителите имат различни нива на достъп до системата, подмодулите са групирани според това за кого е достъпна въпросната функционалност - професори,студенти .

Най-важните качествени характеристики за системата са - отказоустойчивост, производителност, сигурност и използваемост.

За отказоустойчивост на системата се използва подходът с повече от един сървър, на който работи приложението. Така, ако в някой от сървърите настъпи проблем, системата може да продължи работата си без прекъсване, а само с един сървър по-малко. Това е особено важно, когато се провежда кампанията за записване на избираеми предмети,тъй като се предполага че всеки студент ще има еднакъв шанс с останалите за записване на курса, който желае и чиито изисквания удовлетворява. Тогава се очаква, че натоварването на системата ще е по-голямо от обичайното.

Въпреки това тя трябва да осигури, че потребителите могат да я използват надеждно и че техните заявки ще бъдат обработени за не много по-дълго от нормалното време. За запазване на производителността се използва тактиката репликация на модули, в случая на модула, отговорен за основната бизнес логика. Когато имаме няколко независими сървъра, работещи успоредно, системата може да обработи по-голям брой заявки. Броят на репликациите може да варира спрямо натоварването.

Сигурността също е важен фактор, който трябва да се има предвид при проектирането и разработката на системата, тъй като трябва да се осигури, че всички личните данни на потребителите са добре защитени. При регистрация на студентски или лекторски профил се изисква справка на предоставените данни с лични документи. След проверката на администратор, те получават съответните права за ползване на системата, а данните се съхраняват в базата от данни на системата. Освен това специалните данни, като например пароли, се криптират, преди да бъдат изпратени по мрежата. Паролите също така се пазят в хеширан вид в базата данни, за да няма достъп до оригиналната парола. За допълнителна сигурност на данните модулите, отговарящи за главната функционалност на системата, не се свързват директно с базата, а тази връзка се осъществява чрез Database Manager, който предпазва от SQL инжекция и др. За допълнителен контрол на достъпа към сървъра е добавена и защитна стена.

Използваемостта на системата също е приоритет. За да се улеснят потребителите в ориентирането във възможностите на сайта, трябва да се разработи интуитивен потребителски интерфейс. Търсенето на различни събития и курсове да става лесно и бързо, както и използването на онлайн студентската книжка.

Архитектурата на системата гарантира изпълнението на всички изисквания, определени като архитектурни драйвери, както и допълнителни функционални и нефункционални изисквания. Системата е проектирана така, че главните функционалности явно да са обособени в отделни модули и да са осигурени най-важните й качествени характеристики.