



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

Проект

ПО

Софтуерни архитектури и разработка на софтуер

На тема

**CertFIT - Система за генериране и
онлайн проверка на сертификати**

Изготвен от:

Надежда Панделиева – ф.н. 62468

Анита Разпопова - ф.н. 62495

Съдържание

1. Въведение

- 1.1 Обща информация за текущия документ
 - 1.1.1 Предназначение на документа
 - 1.1.2 Описание на използваните структури на архитектурата.
 - 1.1.2.1 Декомпозиция на модулите
 - 1.1.2.2 Структура на процесите
 - 1.1.2.3 Структура на разпределение на работата
 - 1.1.3 Структура на документа
- 1.2 Общи сведения за системата
- 1.3 Разширен терминологичен речник
 - 1.3.1 Списък на софтуерните елементи
 - 1.3.2 Други термини

2. Декомпозиция на модулите

- 2.1 Общ вид
- 2.2 Checking
 - 2.2.1 View
 - 2.2.1.1 CheckAccountID
 - 2.2.1.2 CheckValidity
 - 2.2.1.3 CheckIdentity
 - 2.2.1.4 CheckIssueDate
 - 2.2.1.5 CheckIssueDate
 - 2.2.2 Choose
 - 2.2.2.1 PrintCertificate
- 2.3 Clients
 - 2.3.1 Identification
 - 2.3.2 Accounts
 - 2.3.2.1 Administrator
 - 2.3.2.1.1 Create Templates
 - 2.3.2.1.2 Delete Templates
 - 2.3.2.1.3 Edit Templates
 - 2.3.2.2 Managers
 - 2.3.2.2.1 Generate Certificate
 - 2.3.2.2.2 Search Certificate
 - 2.3.2.2.3 Considerate Certificate
 - 2.3.2.2.4 Download Certificate
 - 2.3.2.2.5 Print Certificate
 - 2.3.2.3 Users
 - 2.3.2.3.1 Access User Certificate
 - 2.3.2.3.2 Search With Unique
 - 2.3.2.3.3 (Un)able Certificate Visibility

- 2.3.2.3.4 Embedding Activities
 - 2.3.3 Диаграма на интерфейса на модулите
 - 2.4 Server
 - 2.4.1 Certificate Editor
 - 2.4.1.1 Inheritance
 - 2.4.1.2 Simple editor
 - 2.4.1.3 Storage and Classification
 - 2.4.2 Hall of Fame
 - 2.4.2.1 Gallery
 - 2.4.3 Security
 - 2.4.3.1 Personal Data
 - 2.4.3.1 Manager Specific Code
 - 2.4.3.1 Certification Deletion
 - 2.4.4 ConnectionWithDB
 - 2.4.5 Диаграма на интерфейса на модулите
 - 2.5 Database
3. Описание на допълнителните структури
 - 3.1 Структура на процесите
 - 3.1.1 Мотивация за избор
 - 3.1.2 Първично представяне
 - 3.1.3 Процес на влизане в системата
 - 3.1.3.1 Описание на елементите и връзките
 - 3.1.4 Процес на издаване на сертификат
 - 3.1.4.1 Описание на елементите и връзките
 - 3.1.5 Процес на търсене на конкретен сертификат
 - 3.1.5.1 Описание на елементите
 - 3.1.6 Процес на използване на Hall of Fame
 - 3.1.6.1 Описание на елементите и връзките
 - 3.2 Структури на разпределение на работата
 - 3.2.1 Мотивация за избор
 - 3.2.2 Първично представяне
 - 3.2.3 Модулите и връзките между тях
4. Архитектурна обосновка
 - 4.1 Минималните характеристики на всеки един сертификат
 - 4.2 Различни роли потребители
 - 4.3 Поддръжка на редактор за сертификати
 - 4.4 Поддръжка на Hall of Fame
 - 4.5 Визуализацията на Hall of Fame
 - 4.6 Сигурност

1. Въведение

1.1 Обща информация за текущия документ

1.1.1 Предназначение на документа

Целта на документа е да представи софтуерната архитектура на системата CertFit.

1.1.2 Описание на използваните структури на архитектурата.

За всяка структура се пояснява какъв точно аспект на системата показва, кои елементи включва и какви са връзките между тях.

1.1.2.1 Декомпозиция на модулите (няма го в темплейта)

Показва обособяването на отделни модули (логически обособени единици) в рамките на системата. Най-общо системата е разделена на модулите Client, Checking, Server, DataBase. Checking служи за справка на потребителите. Client идентифицира правата и действията на съответните потребители, както и осъществява идентификация на потребителите при вход в системата. Server се използва за поддръжка, редакция, интерфейс и защита на системата. DataBase съхранява всички данни в системата и предоставя достъп до тях.

1.1.2.2 Структура на процесите

Изобразяват се нагледно някои от важните процеси на системата

1.1.2.3 Структура на разпределение на работата

Разграничава как ще бъдат разпределени за работа отделните модули между отделните екипи хора, участващи в проекта

1.1.3 Структура на документа

- **Въведение** – заема секция 1 от настоящия документ. Описва се целта и структурата на документа (секция 1.1), дават се общи сведения за системата CertFit (секция 1.2) и съдържа терминологичния речник (секция 1.3).

- **Декомпозиция на модулите** – показва обособяването на отделни модули (логически обособени единици) в рамките на системата. Най-общо системата е разделена на модулите Client, Checking, Server, DataBase. Checking служи за справка на потребителите. Client идентифицира правата и действията на

съответните потребители , както и осъществява идентификация на потребителите при вход в системата . Server се използва за поддръжка, редакция, интерфейс и защита на системата. DataBase съхранява всички данни в системата и предоставя достъп до тях.Тази структура е описана в секция 2 на настоящия документ.

- **Допълнителни структури** – в секция 3 на настоящия документ се разглеждат следните структури:
 - **Структура на процесите** – Разглеждат се някои от по-важните процеси в системата. Тази структура е описана в секция 3.1 на настоящия документ.
 - **Структури на разпределение на работата** – Улеснява създаването на софтуерната архитектура и система, като обособи и разпредели отделните модули от работата по съответните екипи, които да се занимават с тях.Тази структура е описана в секция 3.2 на настоящия документ.
- **Архитектурна обосновка** – Мотивира избраната архитектурна конфигурация и се описва как и защо се постигат основните изисквания на системата. Съдържа се в секция 4 на настоящия документ.

1.2 Общи сведения за системата

CertFit – „Система за генериране и онлайн проверка на сертификати“

Основна цел на CertFit е да се генерират и управляват сертификати, удостоверяващи постижения, знания, участия в курсове, конференции, семинари, практики и стажове в дадена компания.

Системата поддържа три различни роли потребители:

- Администратори, които създават, изтриват и редактират шаблони за сертификати.
- Мениджъри, които генерират сертификати от избран вид, търсят, разглеждат, свалят като файл и разпечатват сертификати от даден тип.
- Обикновени потребители, които могат да разглеждат, свалят като файл и разпечатват техните сертификати, взимат HTML код за вграждане(embedding) на

техен сертификат, търсят сертификат в системата по уникалния му номер, забраняват или разрешават публична видимост на своя сертификат, но не могат да го изтрият.

1.3 Терминологичен речник

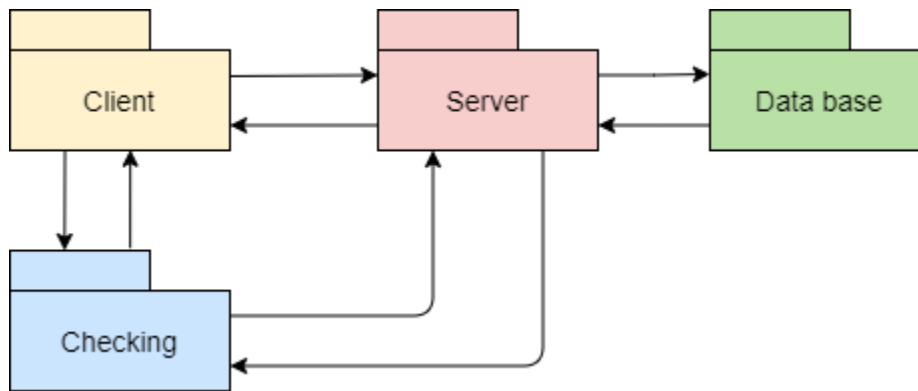
1.3.1 Специфични термини

- **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.
- **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.
- **Приложение (application)** - софтуер, предназначен да помогне на потребителя да извърши определена задача.
- **Интерфейс (interface)** - споделена граница, между която два отделни компонента на компютърна система си обменят информация.
- **Модул** – логически обособена структурна единица.
- **Процес** - съвкупност от стъпки, която изгражда логическо действие и стига определена.

- **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“.

2.Декомпозиция на модули

2.1. Общ вид



2.2. Checking

Този модул служи за справка на клиента. Чрез него се дава достъп до другите подмодули:

2.2.1. View

2.2.1.1. CheckAccountID - съдържа информация за специфичния код на всеки сертификат

2.2.1.2. CheckValidity - съдържа информация за валидността на всеки сертификат

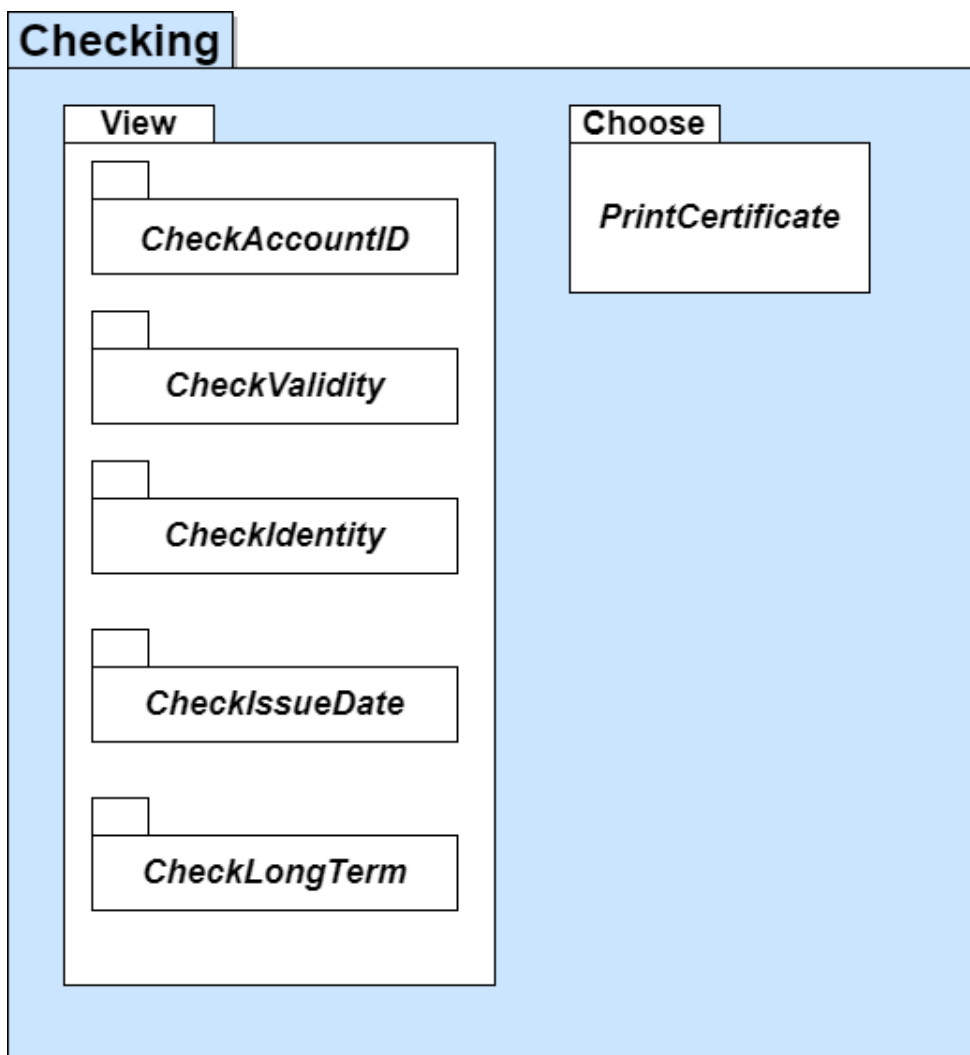
2.2.1.3. CheckIdentity - съдържа информация за идентичността на всеки сертификат

2.2.1.4. CheckIssueDate - съдържа информация за датата на издаване на всеки сертификат

2.2.1.5. CheckLongTerm - съдържа информация за дългосрочността на всеки сертификат

2.2.2. Choose

2.2.2.1 PrintCertificate - всеки потребител може да вземе и да разпечата притежавания от него сертификат



2.3. Clients

Този модул има за цел да представи интерфейс за потребителите и техните права.

2.3.1. Registration - съдържа обща информация за регистрацията на потребителя чрез лични данни. Администраторите и мениджърите имат генерирани профили с вече направена регистрация.

2.3.2. Login

Тук се осъществява идентификацията на потребителя при вход в системата. Проверява се легитимността на потребителя.

2.3.2.1 For Administrator - отделен вход за администратор

2.3.2.2 For Manager- отделен вход за мениджърите

2.3.2.3 For User- отделен вход за обикновен потребител

2.3.3. Accounts

2.3.3.1. Administrator

- 2.3.3.1.1. Create Templates - създава шаблони за сертификати
- 2.3.3.1.2 - Delete Templates - изтрива шаблони за сертификати
- 2.3.3.1.3 - Edit Templates - редактира шаблони за сертификати

2.3.3.2. Managers

2.3.3.2.1 Generate Certificate - Мениджърите могат да генерират сертификати от избран вид.

Също така имат право:

2.3.3.2.2 Search Certificate - да търсят сертификати.

2.3.3.2.3 Considerate Certificate - да разглеждат сертификати.

2.3.3.2.4 Download Certificate - да свалят сертификати като файл.

2.3.3.2.5 Print Certificate - да принтират сертификатите от даден тип.

2.3.3.3. Users

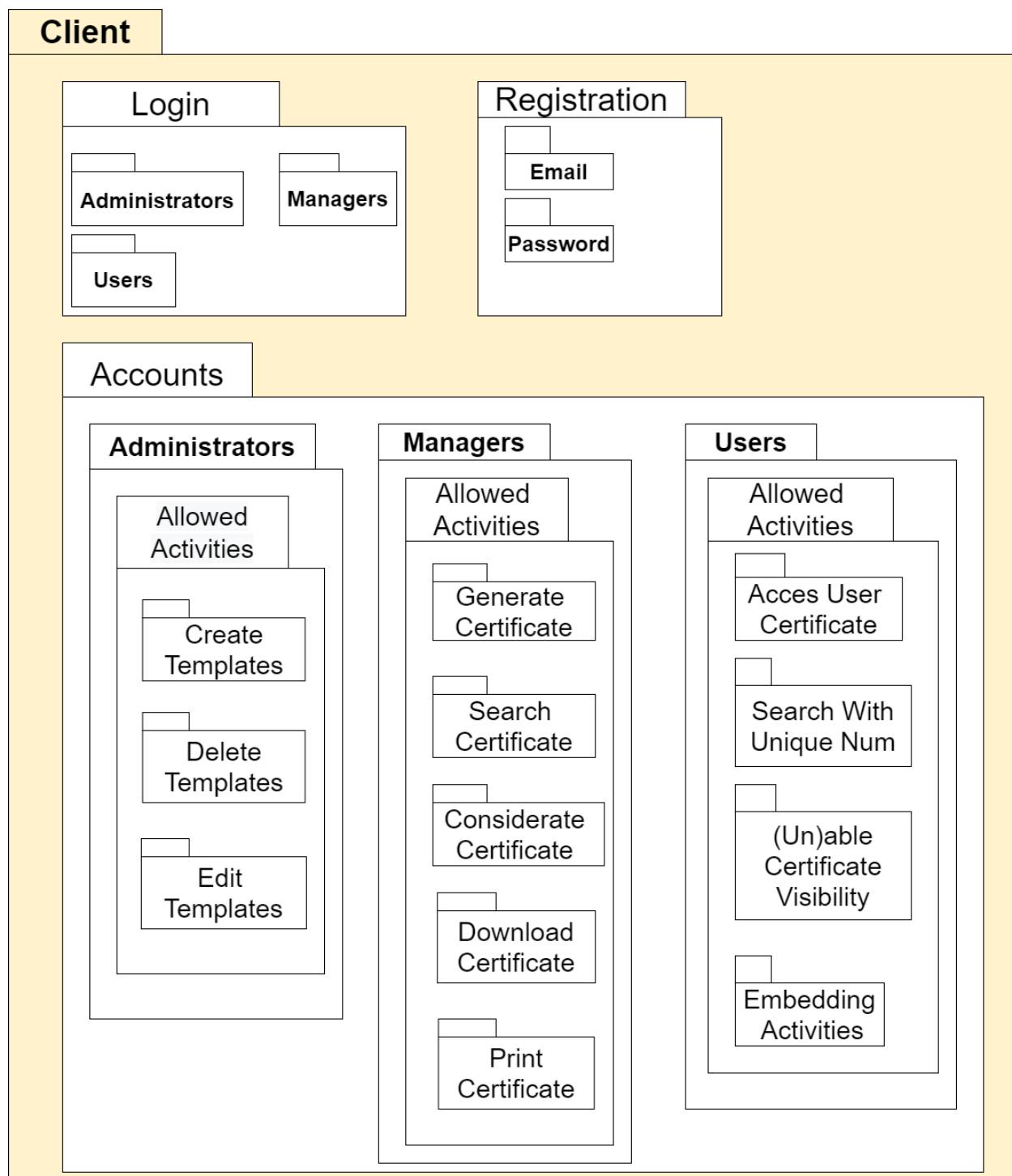
2.3.3.3.1 - Access User Certificate - потребителите могат да разглеждат, свалят като файл и разпечатват техните сертификати

2.3.3.3.2 - Search With Unique - могат да търсят сертификати в системата по уникален номер

2.3.3.3.3 - (Un)able Certificate Visibility - могат да забраняват или разрешават публичната видимост на свой сертификат (но не и да изтриват).

2.3.3.3.4 Embedding Activities - имат право да взимат HTML код за вграждане (embedding) на техен сертификат

2.3.5. Диаграма на интерфейса на модулите



2.4. Server

В този модул се идентифицира поддръжката, редакцията, интерфейса и защитата на системата.

2.4.1. Certificate Editor

2.4.1.1. Inheritance - Клониране/Наследяване на шаблони за

получаване на нови типове сертификати, като това може да бъде използвано за по-лесно изготвяне на нов сертификат, тъй като можем да използваме нещо вече готово и да го подобрим или променим въз основа на нашите искания

2.4.1.2. Simple editor - може да бъде използван при грешки или несъответствия с нашите желания

2.4.1.3. Storage and Classification - Съхраняване, класифициране и разглеждане на графични изображения , които могат да се вграждат в шаблоните са полезни с това, че ще имаме голям избор и разнообразие при изготвяне на нови сертификати.

2.4.2. Hall of Fame

2.3.2.1. Gallery - галерия, в която се виждат сертификатите подредени по ранг, оценка, тип, като това може да бъде полезно както за потребителите, които могат да следят постоянно конкуренцията във всяка отделна сфера, така и на работодателите, които бързо и лесно могат да сравнят

2.4.3 Security

Този подмодул е предназначен за защита на данните от атаки, неоторизиран достъп и всичко свързано с надеждността на системата.

2.4.3.1. Personal Data - голямото количество лични данни като име, ЕГН, адреси и т.н , които трябва да са кодирани и защитени.

2.4.3.2. Manager Specific Code - Чрез ограничение на достъпа, мениджърите ще могат да достъпват сертификатите чрез специфичен код , а не чрез личните данни на потребителите

2.4.3.3. Certification Deletion - Изтриването на сертификат е изпълнение , което ще минава през множество проверки, които ще следят за нерегламентирани права или команди

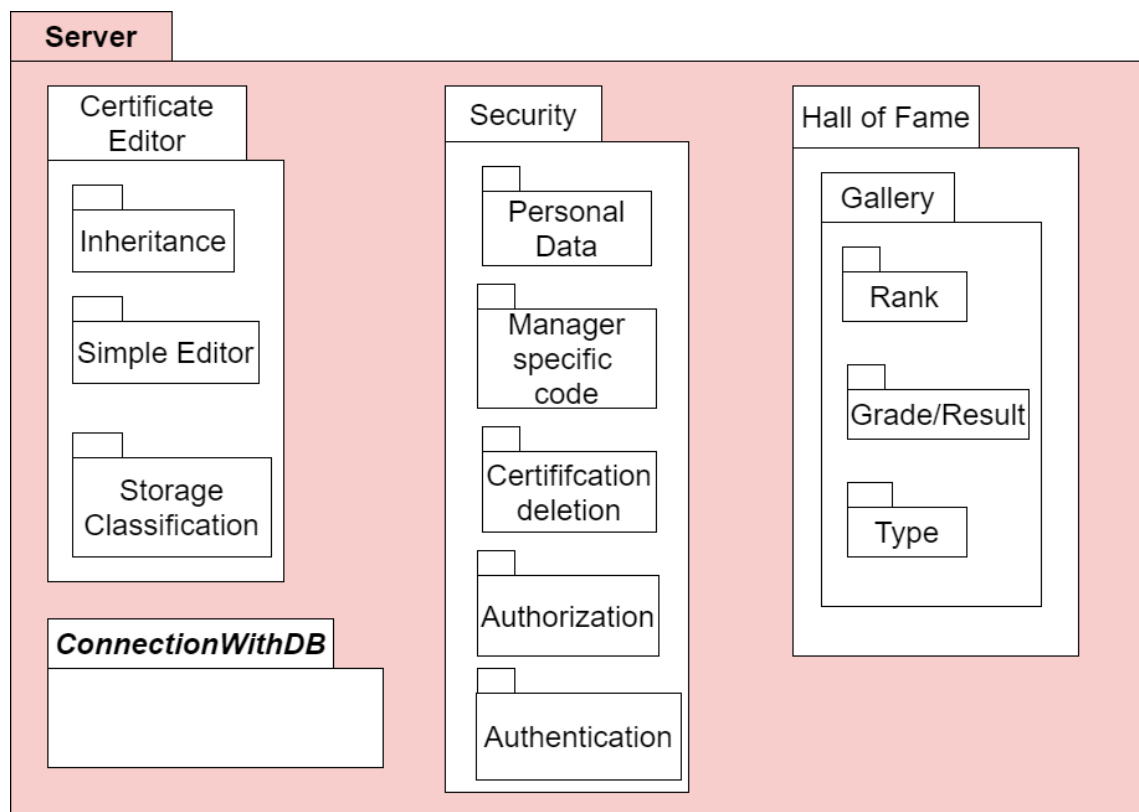
2.4.3.4 Authentication - Осъществява се проверка дали потребителят е този, за когото се представя (извършва се чрез потребителско име и парола).

2.4.3.5 - Authorization - Проверява се дали даден потребител има правата да достъпи дадена функционалност.

2.4.4. ConnectionWithDB

Отговаря за връзката с базата данни, като при възникване на проблем, се изпраща известие до съответната потребителска група.

2.4.5. Диаграма на интерфейса на модулите

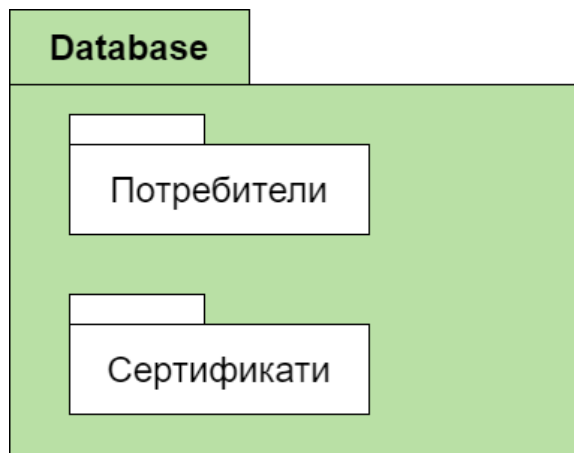


2.5. Database

Този модул се грижи за съхраняването и извличане на данни от системата.

Базата изпълнява заявки в конкретната реализация (SQL, Oracle, ...).

Тя съхранява данните за потребителите, сертификатите и т.н. така, че да са по-бързо достъпни.



3. Описание на допълнителните структури

3.1. Структура на процесите

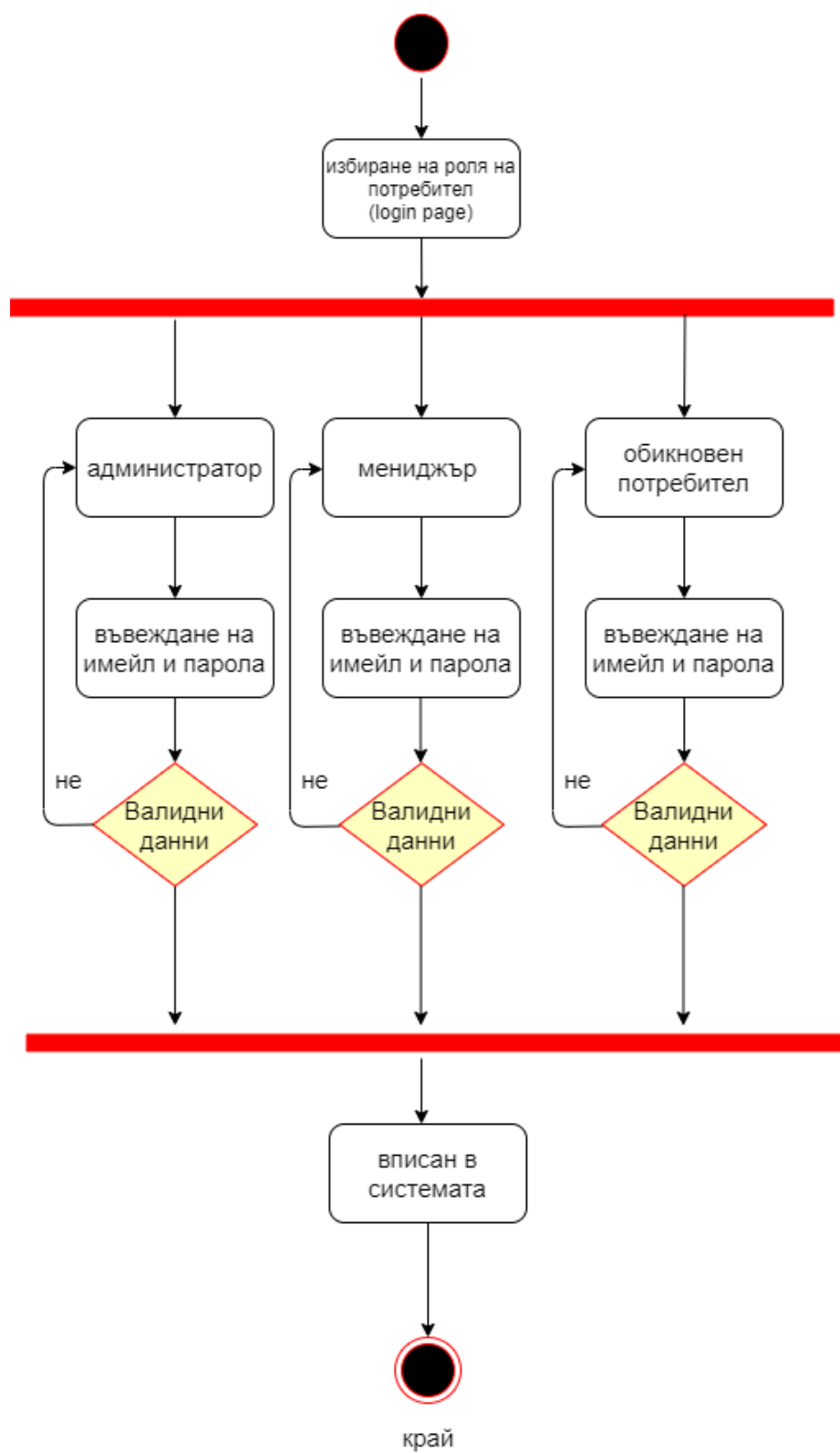
3.1.1 Мотивация за избор

Една от основните архитектурни структури, без които системата би била трудна за проектиране, е структурата на процесите, чиито елементи представляват компоненти, появяващи се по време на изпълнението. Тъй като системата е интерактивна – огромна част от нейната функционалност е базирана на взаимодействие с обикновения потребител. В системата взаимодействието на потребителя се представя чрез набор от функционалности, до които той има достъп – разглеждане и сваляне като файл на техен сертификат, търсене на сертификат по уникален номер, забраняване/разрешаване на публичната видимост на сертификат. Затова тази структура би дала добър поглед над основните взаимодействия на процесите в системата заедно със споделените ресурси, които се ползват – как се променя информацията в базата данни според добавянето на нов сертификат.

3.1.2 Първично представяне

Тази структура се основава на тези компоненти и на комуникацията между тях. Основните въпроси, на които отговаря са кои са основните изчислителни процеси, какви са ресурсите и как са разпределени данните в системата – кои е променят по време на изпълнение на даден процес, как се запазват и как се променя структурата на системата, докато тя работи.

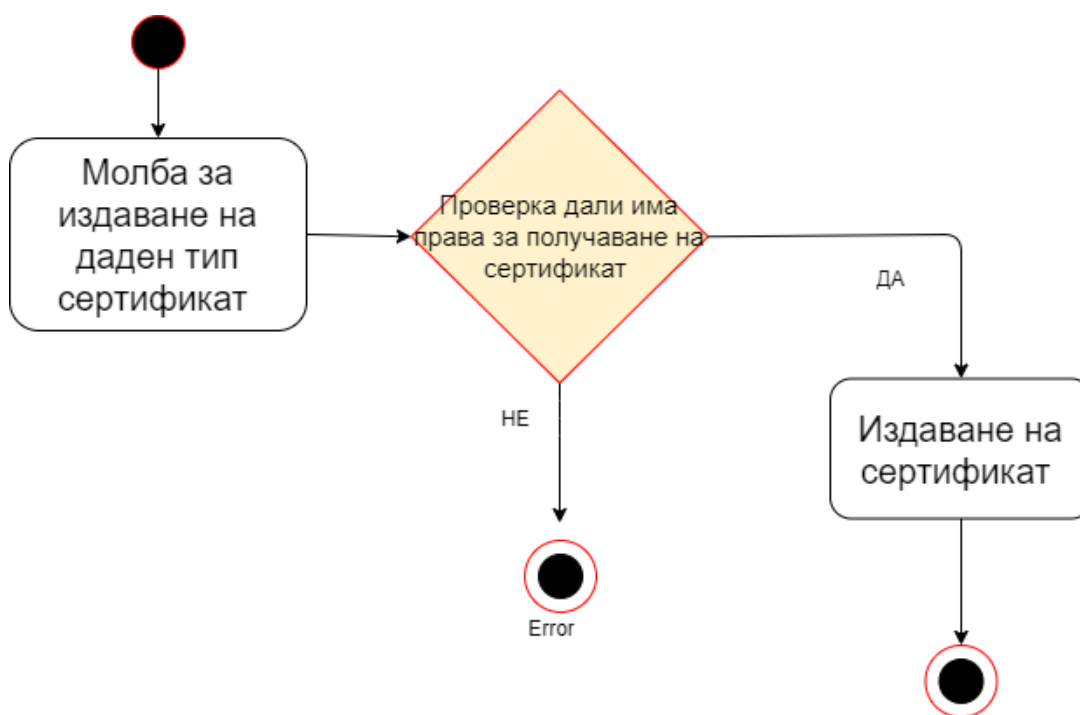
3.1.3 Процес на влизане в системата



3.1.3.1 Описание на елементите и връзките

Даден потребител при искане за вход в системата му се отваря прозорец ,на които трябва да избира като какъв потребител ще влиза - обикновен, администратор или мениджър. След като избере трябва да въведе имейл и парола, които предварително са му били генерирани от системата. Системата прави проверка на входните данни и в следствие, на което ако има грешка го връща в началото , ако ли не е успешно вписан в системата и може да използва предоставените му функционалности според ролята му на потребител.

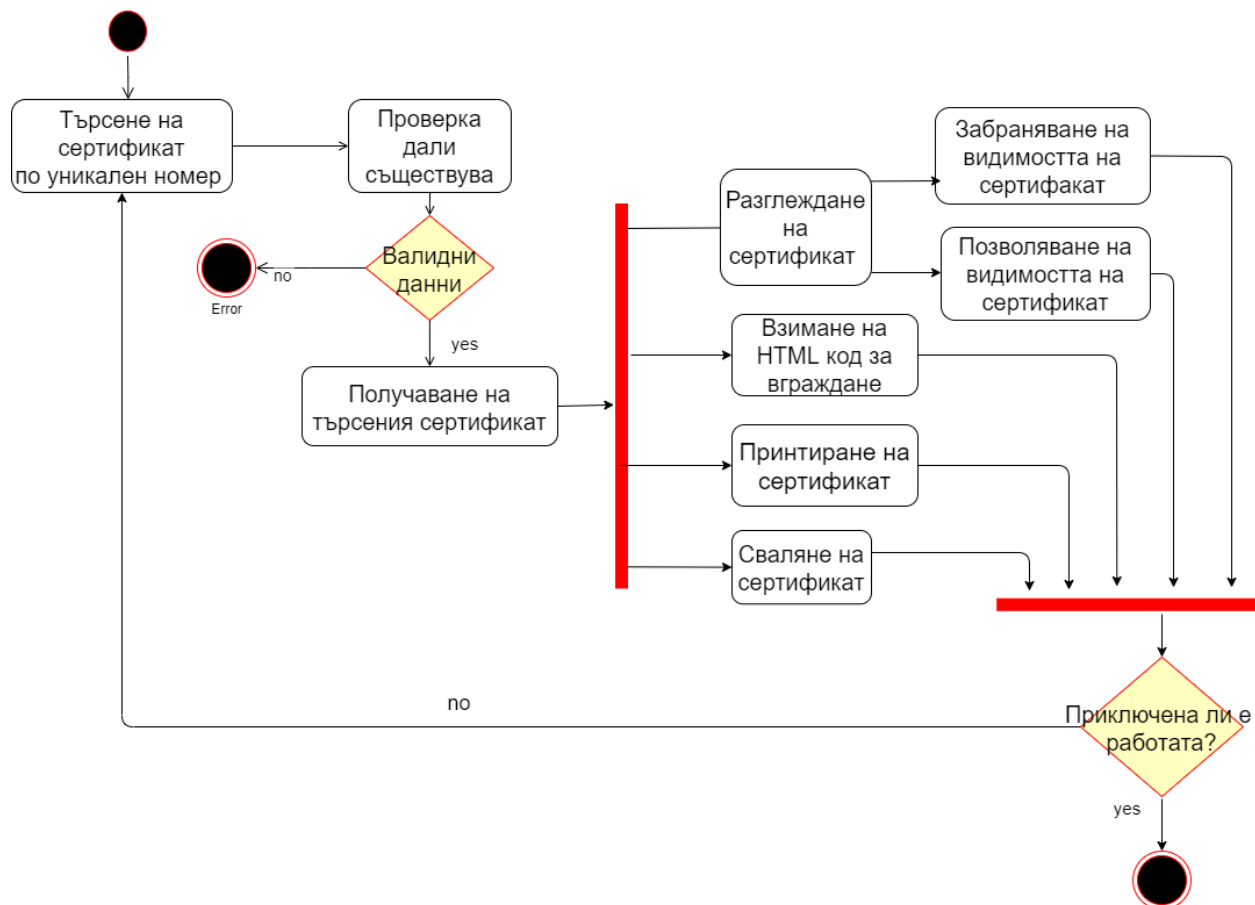
3.1.4 Процес на издаване на сертификат



3.1.4.1 Описание на елементите и връзките

Клиентът подава молба за издаването на сертификата в случай, че е одобрен - при предприемане на тази стъпка от страна на клиента, системата трябва първо да провери дали подателя има право да изисква изкарването на конкретния сертификат. Ако има право сертификата му се генерира и получава сертификат със id код. С този id код потребителят по-късно може да го достъпва. Ако няма системата му извежда съобщение за грешка.

3.1.5 Процес на търсене на конкретен сертификат



3.1.5.1 Описание на елементите

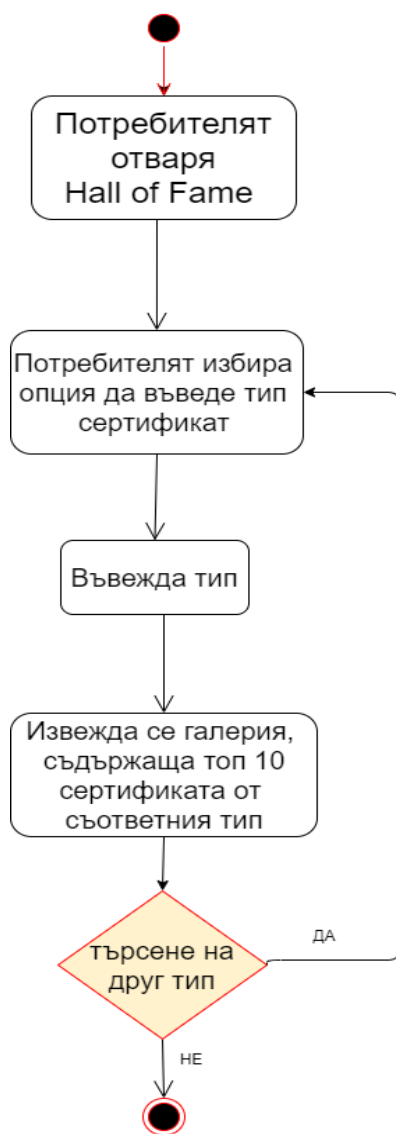
Всеки потребител, който има предоставен уникален номер, има възможност да търси конкретен сертификат. На потребителя се отваря поле за въвеждане, което го подканва да въведе уникалния номер. Ако той е въведен правилно, системата ще намери и изведе търсения сертификат, ако не е въведен правилно ще се изведе грешка.

След като бъде предоставен сертификатът на потребителя той има право на избор какво да прави с него - дали да го разглежда, вземе като HTML код за вграждане, принтира или сваля. Ако избере режим на разглеждане, той има правото да зададе видимостта на своя сертификат, дали да бъде видим само за него или и други потребители да виждат сертификата му, когато влезнат в Hall of Fame(ако той е в класациите). Ако това е всичко, което е искал да направи, потребителят, може да излезе от системата или да се върне към началото си на избор.

Също така има право да иска HTML код за вграждане (embedding), принтира или сваля сертификата като след това му се предоставя

отново изборът да излезе или да се върне към избора си на работа със сертификата.

3.1.6 Процес на използване на Hall of Fame



3.1.6.1 Описание на елементите и връзките

CertFIT поддържа т. нар. "Hall of Fame" за типове сертификати, които съдържат числова оценка. Всеки потребител може да отвори "Hall of Fame" и има възможност да въведе типа сертификат, който го интересува. Тогава системата извежда топ 10 сертификата от въведения тип с най-добри резултати. Това се очаква да бъде полезно както за потребителите, които искат да следят конкуренцията в интересувашата ги сфера така и на работодателите, които могат да добият представа за уменията на кандидатите им чрез предоставените

сертификати. Във всеки един момент може да настъпят промени в сертификатите от дадена галерия, което може да доведе до заблуда и неразбирателство. По тази причина визуализацията на “Hall of Fame” се обновява в реално време, за да поддържа актуална и коректна информация.

3.2. Структури на разпределение на работата

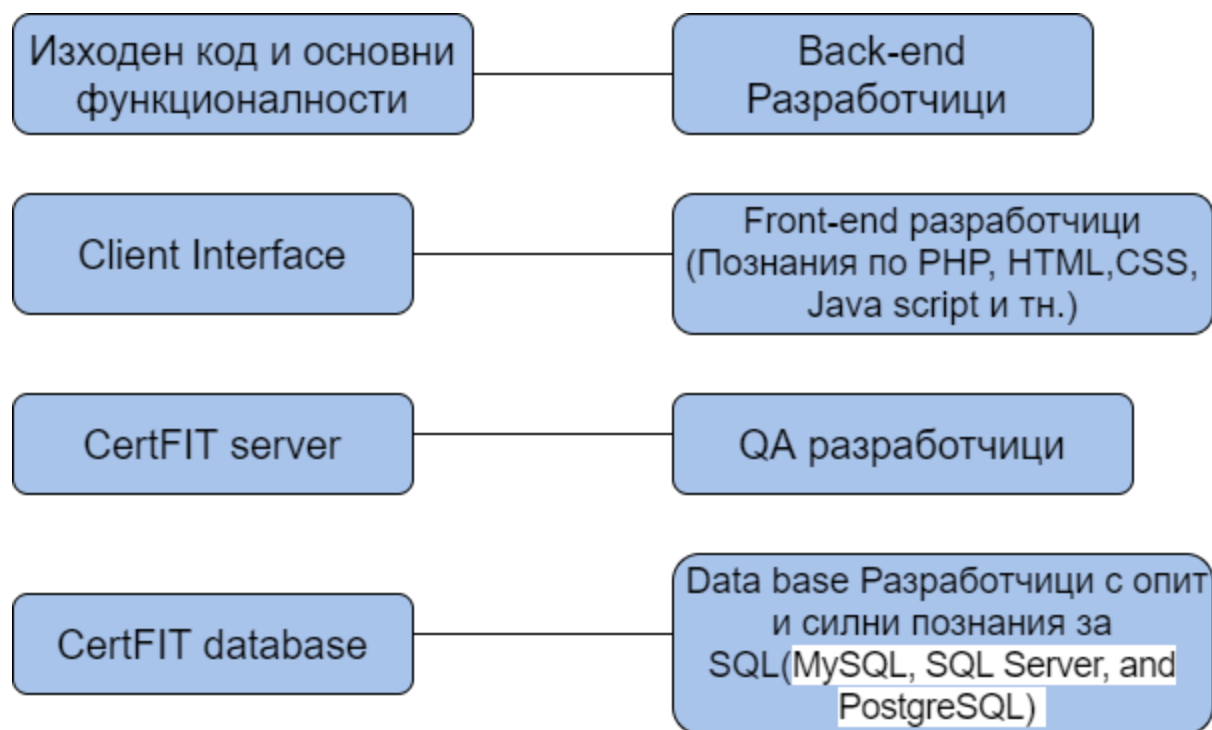
3.2.1 Мотивация за избор

Избрахме тази структура, заради това, че представлява полезен посочен за разграничаване на задачите, включително и от гледна точка на това, че намалява шанса за репликация на грешката – ако проектът не е разпределен има голям шанс да се появят грешки в данните. Тази структура е необходима за правилно разпределяне на работа по екипите и би била полезна на разработчиците да се ориентират към кой екип да се обърнат при въпроси.

3.2.2 Първично представяне

Основна цел на тази структура е да улесни създаването на софтуерната архитектура и система, като обособи и разпредели отделните модули от работата по съответните екипи, които да се занимават с тях. Под екипи се разбира не конкретен списък от хора, а виртуална група от хора, които са групирани на база сходни умения и качества в дадена област, които имат подходящ опит и знания за извършване на някаква конкретна част от реализацията на крайния продукт – например отделни групи хора при разработката биват мениджъри, администратори, системен администратор и т.н.

Основна задача е на архитекта на софтуерната архитектура да подбере хората с нужни качества, които да съответстват на неговите критерии за извършване на коректна и целенасочена работа, което ще намали рисковете и шансовете за неуспех.



3.2.3 Модулите и връзките между тях

Задача за разработване на изходния код на системата, включваща основната функционалност, изисквания за бързодействие, производителност е задача на екипа от Back-end разработчици.

Задачата за разработване на модула Client Interface, който има за цел да представи интерфейс за потребителите и техните права, ще бъде на екипа разработчици, които имат добри познания на HTML, PHP, JavaScript, CSS. Хората в този екип трябва да имат предишен опит със създаването на добре поддържан сайт, който да гарантира на потребителя лесна работа със системата, добре позиционирани бутони и интерфейс, който да им осигурява нужното обяснение, ако имат проблем с дадена функционалност (секция help/short tutorial).

В модулет CertFIT server се идентифицира поддръжката, редакцията, интерфейса и защитата на системата. Тази задача ще бъде поета от QA програмисти, които се занимават с изглаждането на автоматичните тестове, които проверяват постоянно дали системата работи изрядно. В случай на нужда се изпраща информация към екипа на програмистите при нужда за промяна и редакция.

Последният модул е Database – той се грижи за съхраняването и извличането на данни от системата. След като базата изпълнява заявки в конкретната реализация (SQL, Oracle и други) за изработка на съответния модул ще са необходими Database разработчици с опит и силни познания за SQL. След първоначалната реализация на базата данни тя трябва да бъде поддържана непрекъснато.

4. Архитектурна обосновка

4.1. Архитектурни драйвери:

Функционални изисквания:

4.1.1. Минималните характеристики на всеки един сертификат

- тип/шаблон,
- уникален номер
- получател
- издател
- дата на издаване

Това са ключови данни необходими за съществуване на онлайн сертификат. Уникалният номер ще е специфичен код/ID, идентифициращ валидността на сертификата и неговата идентичност. Издателите ще бъдат проверявани в зависимост от типа на документа, дали притежават правата за издаване на такъв документ.

4.1.2. Различни роли потребители

А. Администратори: могат да създават, изтриват и редактират шаблони за сертификати.

В. Мениджъри, които имат следните възможности:

- Генерират сертификати от избран вид
- Търсят, разглеждат, свалят като файл и разпечатват сертификати от даден тип

С. Обикновени потребители, които могат да:

- Разглеждат, свалят като файл и разпечатват техните сертификати
- Взимат HTML код за вграждане (embedding) на техен сертификат
- Търсят сертификат в системата по уникалния му номер
- Забраняват/разрешават публичната видимост на свой сертификат (но не и да го изтрият)

Поддръжката е от изключителна важност за всякакви технически неизправности, които се появяват по време на работния процес. Много грешки и неизправности биха повлияли на сигурността, надеждността и цялостната изправност на системата.

Администраторите са главните потребители. Те могат да създават сертификатите, да изтриват при необходимост или грешки и да редактират при подобряване на оценката или степента на сертификата.

Мениджърите са посредници. Те са връзката между обикновените потребители и администраторите. Те получават сертификатите от администраторите и ги предоставят на обикновените потребители. Те нямат право да създават, редактират или изтриват сертификати, а единствено да ги достъпват.

Обикновените потребители нямат право да създават, изтриват или редактират сертификат. Имат право да търсят сертификат в системата (произволен) на така наречен „празен потребител“ за информация, или в личния му профил, за вече получен сертификат.

4.1.3. Поддръжка на редактор за сертификати

За по-лесно изготвяне на нов сертификат може да използваме клонирани/наследени шаблони за получаване на нови типове сертификати. Това ни позволява да използваме нещо готово и да го подобрим въз основа на нашите изисквания.

При грешки и несъответствия с нашите искания можем да използваме вградения прост графичен редактор.

Съхраняване, класифициране и разглеждане на графични изображения, които могат да се вграждат в шаблоните (т.е. поддържа се библиотека от фонове, логота, разделители и др. изображения) са полезни с това, че ще имаме голям избор и разнообразие при изготвяне на нови сертификати.

4.1.4 Поддръжка на „Hall of Fame”

CertFIT трябва да поддържа т.нар. „Hall of fame” за типове сертификати, които съдържат числова оценка/резултат. Това представлява галерия, в която се виждат Топ 10 сертификата от даден тип. Това се очаква да бъде полезно както за потребителите, които искат да следят постоянно конкуренцията в интересувашата ги сфера, така и на работодателите, които бързо и лесно могат да сравнят получените кандидатури на фона на всички останали и по този начин да направят своя избор.

Нефункционални изисквания:

4.1.5 Визуализацията на „Hall of Fame”

Визуализацията на Hall of Fame да се обновява в реално време е много важна функционалност за системата CertFIT тъй като във всеки един момент може да настъпят промени в дадената галерия, които могат да доведат до заблуда и неразбирателство, а по този начин всяко обновяване на нашата галерия ще бъде достъпно до всеки нуждаещ се от конкретната информация в необходимото време.

За да се достигне това високо ниво на производителност ще използваме определени тактики за производителност. Целта на тактиките за производителност е да се постигне реакция от страна на системата на зададено събитие в рамките на определени времеви изисквания. Възможно ресурсите, заети в обработката на заявките да консумират даденото ни време или системата да блокира в случай на съревнование за ресурси и т.н. - това води до забавяне във времето за реагиране на системата. Тактиката, която трябва да бъде използвана в нашата система е “управление на

ресурсите” - ще се извършва паралелна работа. Промените в базата данни със сертификати ще се отразяват веднага на списъците от Hall of fame паралелно със заявките за визуализиране на топ сертификатите от съответния тип сертификат в платформата и ще води до презареждане на страницата със сертификатите.

Друга необходима тактика е Активен излишък - това е тактика за изправност с цел отстраняване на откази. Компонентите на системата се дублират и се поддържат в едно и също състояние и се използва резултата само от единия от компонентите - активния. Така системата винаги е готова за бърз отговор на съответната заявка дори и при срив.

4.1.6 Сигурност

CertFIT има също големи изисквания към сигурността: невъзможност за нерегламентиран достъп до сертификати, подмяна на съдържанието или изтриването на сертификати, нерегламентирано издаване/генериране на сертификати и т.н.

Сигурността е основна качествена характеристика в CertFIT , поради голямото количество лични данни – име, ЕГН, адреси и т.н в сертификатите. За тази цел ще бъдат предприети няколко тактики за сигурност с цел усвояване на атаки. На първо място всеки потребител ще има регистрация - там ще се съдържат неговите лични данни и неговата роля (обикновен потребител, мениджър, администратор) и при влизане в системата ще се прави автентикация на потребителите чрез имейл и парола.

Втора тактика необходима за поддържането на сигурността е оторизацията на потребителите. Ще се извършва проверка за това дали потребителят има достъп до определени ресурси. Мениджърите ще имат ограничен достъп, като те няма да могат да достъпват сертификатите чрез личните данни, а чрез специфичен сертификационен код. Обикновените потребители ще могат да достъпват произволни сертификати (с цели да проверят как изглеждат/какво представляват) само на „празни профили“. Те ще могат да редактират и достъпват единствено своя личен профил.

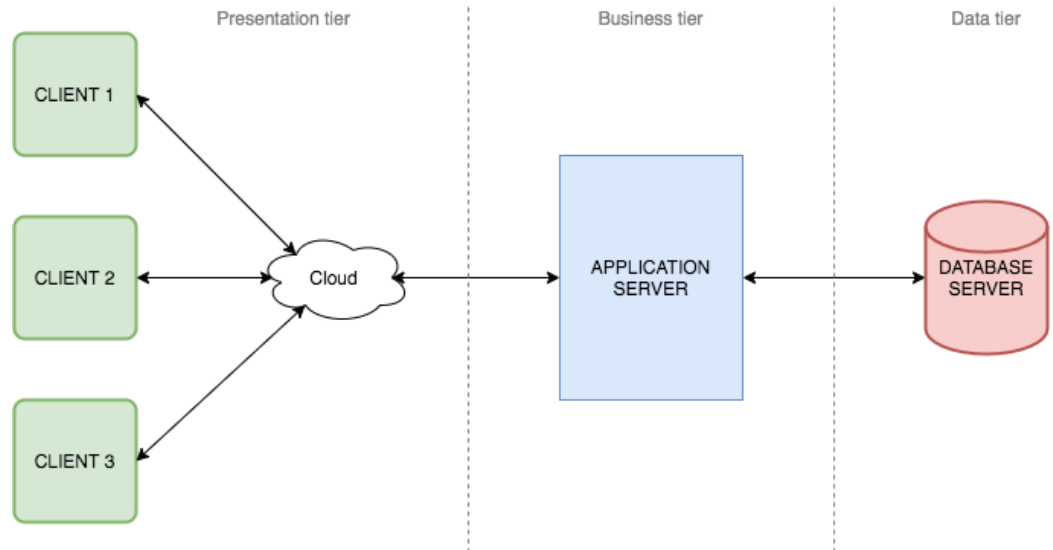
И не на последно място за конфиденциалност на данните ще се използва криптиране на личните данни от сертификатите в постоянната памет.

4.2 Архитектурен стил - Three tier client-server

Според архитектурните драйвери на системата ни, основни и необходими качествени характеристики са сигурността, производителността и изправността. Избрахме архитектурният стил Three tier client-server, защото той допринася за по-добра сигурност и производителност. Други силни страни на client-server стила са лесното архивиране и възстановяване на данните. Three tier структурата предлага по-висока гъвкавост, поддръжка и т.н. от two tier. Имаме 3 нива:

1.Презентационно ниво, което представлява клиентите.

2. Бизнес ниво, което действа като посредник за частично обработени данни.
3. Ниво от данни, което съдържа сървъра на базата данни.



По-лесно може да се внедри подобрена защита, тъй като клиентските приложения нямат пряк достъп до база данни с профили и сертификати. Възможна е отделна промяна на съответното ниво без да се засягат другите нива на системата. И имаме подобрена цялост на данните, тъй като повреждането на данни от клиентски приложения може да бъде елиминирано чрез предаване на данните в бизнес нивото за проверка.