Техническа Документация

[AprioriTID]

Екип: [Александър Иванов, Борислав Тонев, Радина Петришка]

Дата: [03.06.2020]

# История на техническата документация

**Автори**

| Фак. номер | Име | Контакт (ел. поща) |
| --- | --- | --- |
| 501217031 | Александър Иванов |  |
| 501217049 | Борислав Тонев |  |
| 501217014 | Радина Петришка |  |

**История на версиите**

| Версия | Дата | Автор | Описание |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

# 

# Съдържание

[1 Въведение 4](#_Toc446758084)

[1.1 Цели на системата 4](#_Toc446758085)

[1.2 Обхват на системата 4](#_Toc446758086)

[2 Спецификация на изискванията 4](#_Toc446758087)

[2.1 Системни изисквания 4](#_Toc446758088)

[2.1.1 Функционални изисквания 4](#_Toc446758089)

[2.1.2 Нефункционални изисквания 4](#_Toc446758090)

[2.2 Изследване на потребителските изисквания 4](#_Toc446758091)

[2.2.1 Потребителски истории 4](#_Toc446758092)

[2.2.2 Критерии за приемане на потребителски истории 5](#_Toc446758093)

[3 Концептуален дизайн на системата 5](#_Toc446758094)

[3.1 Софтуерна архитектура на система 5](#_Toc446758095)

[3.2 Декомпозиция на системата на модули 5](#_Toc446758096)

[3.3 Концептуален модел на БД 5](#_Toc446758097)

[4 Ресурси 5](#_Toc446758098)

[4.1 Акроними 5](#_Toc446758099)

[4.2 Други 5](#_Toc446758100)

# Въведение

## Цели на системата

Desktop-базирана система за анализ онлайн курсове. Всеки администратор е нужно да разполага с валиден баркод, за да влезе в системата. Целта на система е реализира алгоритъма AprioriTID, чиято цел е да открие елементи, които са най-често срещащите се елементи по зададен критерии от файла. Това са активните потребители, преглеждали най-много файлове, или файлове, които са били преглеждани най-много пъти и др. Системата бива използвана от лица, отговарящи за поддръжката с цел проследяване на информацията за активните потребители и често посещавани файлове.

Алгоритъмът се извиква като му се подава предварително подготвен файл, съдържащ инфорамация за потребителите – ID и/или ID на файл.

## Обхват на системата

Приложението ще се използва само лица(администратори), които разполагат с валиден баркод. Основните функционалности, които са заложени в управлението на системата – извеждане на информация за курсовете и файловете, тяхното преглеждане, статистика за честотата на преглеждането им от потребителите, сортиране им по различни критерии.

В обхвата на системата влизат част от данните на лог файла logs\_BCS37\_20181103.csv.

От лог файлът, за да изпълним зададения алгоритъм, взимаме само записите, които имат event\_context=’File’ или event\_context =’Course’, event\_name=’Course module viewed’ или event\_name=’Course viewed’, IP address, Time, Description. От тези записи в последствие остават само необходимите ни полета – ID на модул и ID на потребител от колоната Description,както и колоните име на курса/файла, IP на потребител и Дата за представяне на данните в разбираем вид след изпъление на алгоритъма.

# Спецификация на изискванията

## Системни изисквания

### Функционални изисквания

| **Номер** | **Изисквания** | **Приоритет** |
| --- | --- | --- |
| 1 | Избор на подходящ лог файл за алгоритъма. | Висок |
| 2 | Подбор на нужните данните – взимаме колоните от лог файла, който ще изпозваме в алгоритъма. | Висок |
| 3 | Предварителна обработка и трансформация на данните – изчистване на данните, който не отговарят на критериите, заложени в обхвата на системата. | Висок |
| 4 | Анализ на данните – прилагане на алгоритъма и краен резултат. | Висок |
| 5 | Оценка на крайния резултат – данните да се представят в подходящ изходен формат. | Среден |
| 6 | Логването в система да става чрез баркод. | Висок |
| 7 | Да сортирам файловете в системата по брой преглеждания. | Висок |
| 8 | Извеждане на най-честно преглежданите файлове. | Среден |
| 9 | Възможност за намиране на файл по ID. | Среден |
| 10 | Възможност за преглеждане на всички файлове/курсове. | Нисък |
| 11 | Достъп до системата имат само оторизирани лица(администратори) с валиден баркод. | Висок |

### Нефункционални изисквания

| **Номер** | **Изисквания** | **Приоритет** |
| --- | --- | --- |
| 1 | Desktop приложение. | Висок |
| 2 | Приложението да е реализирано на програмния език Java. | Висок |
| 3 | Да се използва CSS и .fxml файлове за визуализиране на потребителския интерфейс. | Висок |

## Изследване на потребителските изисквания

### Потребителски истории

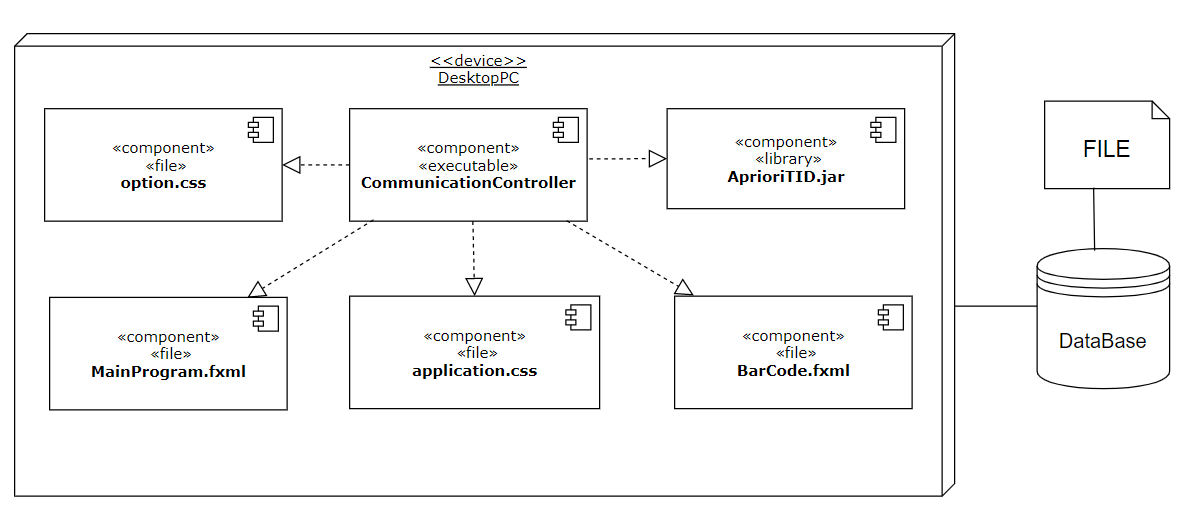
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Номер на потр. история** | **Като** | **аз искам да** | **така, че** | **точки на историята** |
| 1 | Администратор | се намери конкретен курс/файл по въведено от мен ID | да видя цялата информация за ползването му и от кого | 21 |
| 2 | Администратор | се извежда пълния списъс на наличните файлове/курсове в системата | да видя всички ресурси в системата | 13 |
| 3 | Администратор | се изведат най-често посетените файлове от потребител с ID, въведено от мен | да видя от какво се интересува най-много конкретен потребител | 21 |
| 4 | Администратор | се изведе информация за най-честите посетители в системата | да разбера кои потребители са преглеждали най-много файлове/курсове | 21 |
| 5 | Администратор | се изведе информация за най-често посетените файловете | да разбера от какво се интересуват и какво предпочитат потребителите | 21 |
| 6 | Администратор | се индентифицирам чрез баркод | да мога вляза успешно в системата | 34 |
| 7 | Администратор | да се избере подходящ log файл | да мога да приложа данните в конкретния алгоритъм |  |
| 8 | Администратор | се извлекат данните(колоните) от log файла, които са ни нужни | да бъдат лесно и бързо обработени в последствие | 34 |
| 9 | Администратор | се направи предварителна обработка | да се премахнат шумът (некоректните) и непълните данни | 55 |
| 10 | Администратор | данните да се кодират с числови стойности | да се подадат коректни, точни и подходящи за алгорима данни | 34 |
| 11 | Администратор | да получа анализ на данните | да мога да водя статистика за информацията в системата | 21 |
| 12 | Администратор | да получа данните в разбираем текстови формат | да мога лесно и недвусмислено да разбера крайният резултат от анализа на данните | 21 |

### Критерии за приемане на потребителски истории

|  |  |
| --- | --- |
| **Номер на потр. история** | Критерии за приемане |
| 1 | Да се изведе пълната информация за намерия курс/файл. |
| 2 | Да се изведе на екрана информация за всички налични курсове/файлове в системата. |
| 3 | Да се изведе датата и цялата информация на посещения на потребител с конкретно ID. |
| 4 | Да се изведе на екрана ID-тата и IP-тата на най-честите посетители в системата. |
| 5 | Да се изведе на екрана най-често преглежданите файлове в системата. |
| 6 | В система се влиза само чрез един единствен баркод. |
| 7 | Файлът трябва да съдържа данни в числов формат, за които да е възможно да се правят изчисления на базата на алгоритъма AprioriTID или други изчислителни алгоритми. |
| 8 | Данните(колоните) да бъдат съхранение във файл с подходящ формат, за да могат да се извлекат лесно след това стойностите, които ще бъдат необходими на алгоритъма. |
| 9 | От данните да останат само тези записи, които отговарят на критерия описан в обхвата на системата. |
| 10 | Данните да са числов формат и да бъдат съхранение във файл, който в последствие ще се подаде на алгоритъма за изчисления. |
| 11 | Да получа данните от резултата на алгоритъма в числов формат за последваща обработка в по-четим вид. |
| 12 | Получените резултати от предходния критерии да бъдат представени по разбираем, лесен за прочитане и анализиране начин на екрана. |

# Концептуален дизайн на системата

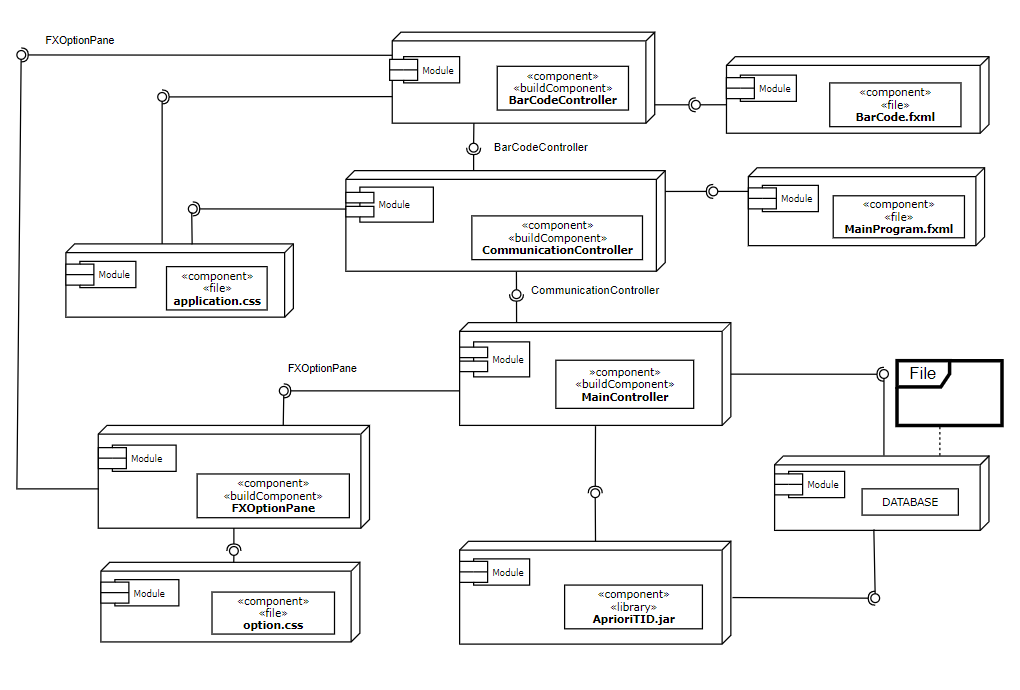
## Софтуерна архитектура на система



*На снимката 1 е представена диаграма на внедряване на системата.*

MainController-a е модулът (звеното), което осигурява връзка между всички компоненти, файлове и библиотеки в системата. От него се стартира програмата и според желанието на администратора се изпълняват останалите модули.

## Декомпозиция на системата на модули



*На снимката 2 е представена компонентната диаграма на системата.*

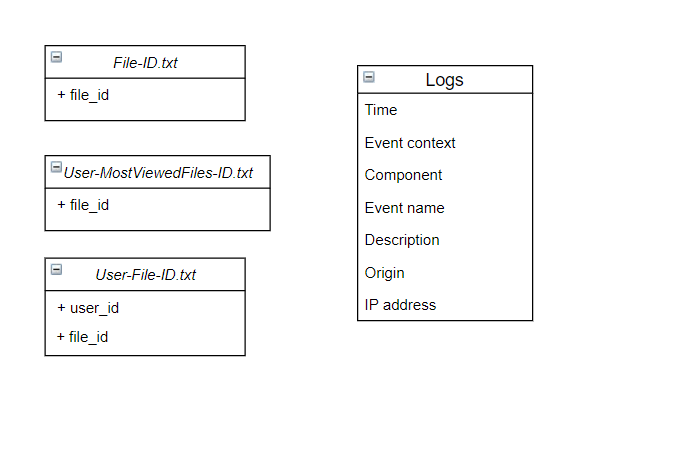
Показани сa компонентите, с които отделните модули си комуникират.

Fxml файл компонентите – MainProgram.fxml, BarCode.fxml, отговаря за дизайна на приложението. Те се грижат за разположението на бутоните, текстовите полета, layout-те и много др.

Контролерите – MainController, CommunicationController, BarCodeController, FXOptionPane отговарят за имплементирането fxml и css компонентите, дизайна и backend логиката на приложението. В MainController-а също така се използва библиотеката AprioriTID.jar и чрез нея имплементираме логиката на използвания алгоритъм – AprioriTID, като след това се правят преобразувания на изходните данни, така че резултатът да се представи по разбираем начин.

Компонентът application.css отговаря за визуалното представяне на интерфейса – тема на приложението, бутони – цвят, размер, рамки на бутоните и на целия интефейс и много др., а option.css за прозорчето, което се появява след натискане на някой от бутоните.

## Концептуален модел на БД



*На снимката 3 е представен ER диаграмата на системата.*

Таблицата Logs представлява данните от лог файла в оригиналния им вид, без никакви обработки – непълни, некоректни, дублиращи или не отговарящи на условията за използване на алгоритъма – AprioriTID.

Данните се обработват – премахват се повтарящите се записи, които не съдържат ид на потребител или ид на курс/файл. Също така се взимат записите, които отговарят на условията поставени в обхвата на системата - event\_context=’File’ или event\_context =’Course’, event\_name=’Course module viewed’ или event\_name=’Course viewed’, IP address, Time, Description. От тези записи в последствие остават само необходимите ни полета – ID на модул и ID на потребител от колоната Description, име на курса/файла, IP на потребител и Дата.

След обработката се записват в User-File-ID.txt, където се съхраняват ид на потребителите и ид на файловете, като сме извлекли така, че да ни дават информация всеки потребител(според ид-то) си кой файл(според ид-то му) е посетил.

От User-File-ID.txt можем да извлечем всички налични файлове според ид-тата им и ги записваме във файл File-ID.txt, които използваме в нашите сторита и подаваме на алгоритъма за анализ.

User-MostViewedFiles-ID.txt съхранява информацията динамично, според въведено от администратора ид на потребител, се записват всички посетени файлове от потребителя в низходящ ред.

# Ресурси

## Акроними

| **Акроним** | **Описание** |
| --- | --- |
| IP | Internet Protocol |
| ID | IDentification |
|  |  |

## Други

<http://www.philippe-fournier-viger.com/spmf/AprioriTID.php>

<http://www.philippe-fournier-viger.com/spmf/index.php?link=download.php>