

Требования к проекту

1 Введение

Проект носит название 2Chat. Должны быть реализованы два модуля – клиентский, и серверный. Много клиентов будут присоединяться к одному серверу, образуя сеть взаимодействия. Пользователи смогут обмениваться анонимными сообщениями на интересующие их темы.

2 Требования пользователя

2.1 Программные интерфейсы

Клиент-серверное взаимодействие будет реализовано через протокол TCP, т.к. необходимо доставлять сообщения без повреждений. Для простой и надежной реализации многопоточной обработки данных на сервере, сервер и клиент будут реализованы на языке Elixir. Графический интерфейс будет реализован на Java с помощью JavaFX.

2.2 Интерфейс пользователя

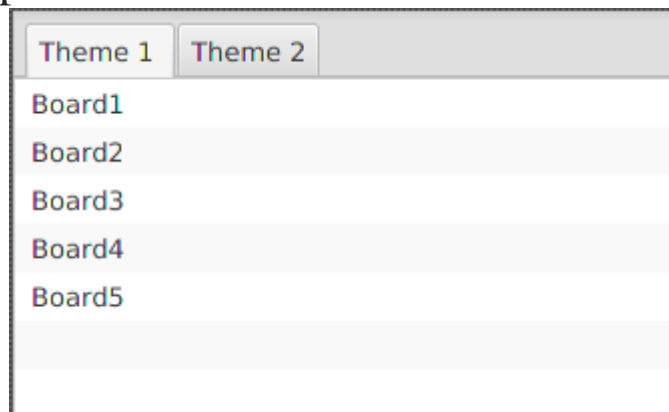


Рис.1 Окно выбора раздела



Рис.2 Окно выбора нити

Return Board Title - Thread Title

Post1

Post2

Post3

Post4

Post5

Your post here

Option button1

Option button2

Option button3

Send Button

Рис.3 Окно нити

2.3 Характеристики пользователей

Пользователи различные настолько, насколько это позволяет количество тематических разделов. Необходима только простейшая компьютерная грамотность.

2.4 Предположения и зависимости

Возможно, передача видео будет тяжело выполняема описанными средствами, поэтому может быть использована сторонняя библиотека.

3 Системные требования

3.1 Функциональные требования

- 1) Все пространство общения будет делиться на крупные тематические разделы. Разделы может создать только администратор, имеющий доступ к терминалу сервера. В каждом разделе можно создавать нити обсуждений, сконцентрированных на некой одной теме. Любой пользователь имеет право создать нить.
- 2) Каждый клиент сможет оставить анонимное сообщение. К сообщению можно прикреплять видео или изображение. Предельные размеры сообщений вместе с прикрепленными данными будут высчитываться в ходе разработки.
- 3) Каждое обычное сообщение является ответом на какое-то другое сообщение. Только сообщение в заголовке нити не является ответом. Серию ответов можно легко проследить в окне чата.
- 4) Нить имеет лимит сообщений, при достижении лимита нить будет удаляться. Лимит для каждого раздела устанавливает администратор.
- 5) Пользователи могут оставлять жалобы на нити, чтобы привлечь внимание администратора к правонарушениям.
- 6) Доступ к терминалу сервера может быть ограничен и защищен паролем. Пользователь с доступом к терминалу сервера является администратором, и может удалять сообщения, нити, разделы и даже сам чат. Он имеет доступ к данным о источниках тех или иных сообщений. Так же он может блокировать источники, если сочтет нужным, например, в случае нарушений правил.
- 7) Относительная анонимность. Администратор при желании сможет вычислить пользователя. Если пользователь не сможет вычислить другого пользователя, значит анонимность соблюдена. Это привлечет много интересных тем.

3.2 Нефункциональные требования

3.2.1 АТРИБУТЫ КАЧЕСТВА

* Гарантия передачи сообщения, т.к. передача сообщений – основная задача чата. Для этого будет использован протокол TCP. Протокол TCP удобен в использовании, популярен, и надежен.

3.2.2 ОГРАНИЧЕНИЯ

* Для каждого пользователя будет создан отдельный поток обработки на сервере. Это легко реализуется на языке Elixir.

* Протокол связи должен быть выделен в отдельный модуль, чтобы его можно было заменить.

3.2.2 ЭТИЧЕСКИЕ ТРЕБОВАНИЯ

* Темы для общения и содержания нитей могут быть любыми, за исключением материалов, нарушающих законы УК РБ.