Software requirements specification (SRS)

Структура документа:

* Introduction
  + Purpose
  + Document conventions
  + Intended Audience and Reading Suggestions
  + Project scope
* Overall Description
  + Product perspective
  + Product features
  + User classes and characteristics
  + Operation environment
  + Design and implementation constraints
  + User documentation
  + Assumptions and dependencies
* System features
  + System feature vClipboard
    - Description and priority
    - Stimulus/Response sequences
    - Functional requirements
  + System feature most.profit.month
    - Description and priority
    - Stimulus/Response sequences
    - Functional requirements
* External interface requirements
  + User interfaces (UIs)
  + Software interfaces
  + Hardware interfaces
  + Communication interfaces
* Non functional requirements
  + Performance requirements
  + Safety requirements
  + Software quality attributes
  + Security requirements

**Introduction \ Purpose.**

Раздел сайта gamesport.com - “История прогнозов” (<https://gamesport.com/ru/forecasts/history>) содержит в себе полную историю прогнозов с Февраля 2020 года.

**Introduction \ Document conventions.**

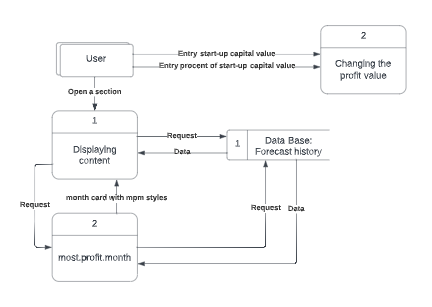
* фича-функция
* user-пользователь
* most.profit.month-функция оформления наиболее выгодного месяца
* vClipboard-функция “умная статистика”
* ЯП- язык программирования
* IDE-
* select\_\_value-определенный год в используемый в функции “умная статистика”
* i\_1 и i\_2-значения полей “стартовый банк” и “процент от банка” в функции “умная статистика”
* БД-база данных

**Overall Description \ Product perspective.**

Главная перспектива используемых фичей в разделе “История прогнозов” - это взаимодействие пользователя и истории прогнозов, что позволяет user’у подробнее ознакомиться с результатами деятельности команды аналитиков. Также привлечь внимание к достижениям аналитиков киберспортивных матчей по средствам most.profit.month.

**Overall Description \ Product features.**

Ниже представлена DFD-диаграмма, которая описывает взаимодействие user’а и системы.



**Overall Description \ User classes and characteristics.**

Основной класс пользователей, которые будут использовать функционал раздела - это посетители сайта, заинтересованные в изучении статистики команды аналитики киберспортивных матчей. При помощи предоставленного функционала пользователь сможет оценить качество аналитики матчей, что позволит заработать доверие со стороны пользователя.

**Overall Description \ Operation environment.**

Продукт должен корректно работать во всех браузерах, в том числе десктопных и мобильных. Код продукта должен компилироваться во всех IDE, поддерживающих веб-разработку.

**Overall Description \ Design and implementation constraints.**

* используемые ЯП: html+css & JS.
* стандарты кодирования принятые в отделе разработки
* стандарты обмена данными заложенные в отделе разработки

**Overall Description \ User documentation.**

Для использования функции пользователю нет необходимости изучения той или иной документации.

**Overall Description \ Assumptions and dependencies.**

Факторы которые могут повлиять на работу функционала раздела “Истории прогнозов”:

* изменения данных в БД (весь функционал полностью завязан на данных которые получаются из БД)

**System features \ System vClipboard.**

Функция “Умная статистика”, уникальный идентификатор - vClipboard (p.s. я не очень JS разработчик, но после изучения JS кода, мне показалась что именно эта функция отвечает за умную статистику, вероятней всего я ошибся (: ).

**System features \ System feature vClipboard \ Description and priority.**

Данная фича дает возможность пользователю ознакомиться с возможной прибылью исходя из его стартового банка и процента денег, который он готов ставить относительно своего банка. Функция рассчитывает точную прибыль пользователя в каждый месяц того или иного года, который он выбрал в выпадающем списке. Приоритет данной функции нулевой, т.к. она уже реализована на сайте.

**System features \ System feature vClipboard \ Stimulus/Response sequences.**

Триггером запуска фичи является ввод значения в одно из полей (или оба) “Стартовый банк” & “Процент от банка”. Впоследствии в карточках месяцев изменяется сумма прибыли в зависимости от введенных значений.

**System features \ System feature vClipboard \ Functional requirements.**

Фича должна ограничивать ввод возможных значений пользователя, чтобы избежать математических противоречий во время расчёта прибыли. Поэтому в поле “Стартовый банк” пользователь может ввести 0<X<100000001 (сто миллионов один), а в поле “Процент от банка” можно ввести 0<X<101 (сто один). При попытке ввести значение не удовлетворяющие неравенству, значение автоматически меняется на удовлетворяющие наиболее близкое к введенному. Значения прибыли, которые получаются в итоге вычислений отображаются в правом верхнем углу в карточке месяца.

**System features \ System most.profit.month.**

Функция “Наиболее доходный месяц”, уникальный идентификатор - most.profit.month.

**System features \ System most.profit.month \ Description and priority.**

Данная фича выделяет наиболее доходный месяц в году, что привлекает внимание пользователя и улучшает общий дизайн раздела. Функция анализирует показатели прибыли каждого месяца в году, после чего сравнивает их между собой и применяет к карточке наиболее доходного месяца соответствующие стили. Приоритет данной функции наиболее высокий (1 из 1).

**System features \ System most.profit.month \ Stimulus/Response sequences.**

Триггером запуска фичи является загрузка страницы пользователем, а также при изменении года в выпадающем списке, т.е. при изменении значения select\_\_value, в появившемся на экране пользователе блоке карточек месяцев, должна быть выделена карточка наиболее доходного месяца. При выборе значения “-” в select\_\_value фича должна найти наиболее выгодный месяц среди 11 последних месяцев, вне зависимости от года.

**System features \ System most.profit.month \ Functional requirements.**

Фича должна корректно взаимодействовать с vClipboard, чтобы в случае изменения значений i\_1 и i\_2, наибольшая месячная прибыль вычислялась верно. В случае, если итоговая месячная прибыль у нескольких месяцев равна и является наибольшей, необходимо применить стили most.profit.month ко всем карточкам этих месяцев.

**External interface requirements.**

Фича most.profit.month будет получать значения прибыли из БД и преобразовывать путем математических формул в необходимые данные. Для корректного отображения на экране пользователя информации будет предоставляться путем оформления при помощи HTML и CSS.

**Non functional requirements.**

* Функция не должна перегружать ресурсы веб-сервера, чтобы избежать нежелательных багов и перегрузок
* Функция должна соответствовать всем протоколам безопасности, чтобы избежать уязвимость веб-сайта.
* Функция должна корректно работать как в десктопных браузерах, так и мобильных.

**Non functional requirements \ Performance requirements.**

Требования к производительности:

* время от момента получения запроса до момента создания полного ответа не должно превышать 200 мсек, при условии одновременной обработки не более 750 запросов;
* объем используемой виртуальной памяти (включая кеш) не должен превышать 0.2 Гбайт, при условии одновременной обработки 750 запросов и не более 10000 открытых сеансов пользователя;
* нагрузка на CPU не должна превышать 5%, при условии не более 5000 одновременно обслуживаемых запросов
* объем памяти для хеширования объектов JS в окне не должен превышать 0.5 Мбайт

**Non functional requirements \ Software quality attributes.**

Требования к качеству кода и метрики которые используются для определения качества кода, а также кол-во кода, который должен быть покрыт тестами и принцип их использования - определяется командой разработчиков, ответственной за исполнение обозреваемого продукта.

**Non functional requirements \ Security requirements.**

Разработанные фичи должны быть реализованы так, чтобы с их помощью не было возможности перегрузить сервер или получить доступ к веб-серверу и БД.