Функции класса Web: имеют тип void, работают в потоках и обращаются к серверу lab, результат записывается в переменную Web.res, возможны следующие значения Web.res:

1. "" - пустая строка, в этом случае функция ещё не закончила своё выполнение.
2. "-1" - проблема на клиенте (сервер lab в принципе такого не возвращает): отсутствие интернета.
3. "0" - успешное выполнение.  
   (если это значение указано в столбце «Возвращает» таблицы ниже)
4. "1" - признак того, что нужно ещё подождать и перезапустить функцию.   
   (если это значение указано в столбце «Возвращает» таблицы ниже)
5. Что-то другое

|  |  |  |  |
| --- | --- | --- | --- |
| Функции класса Web | Требуемые параметры | Назначение | Возвращает |
| create(n) | Вместимость комнаты | Создание комнаты, в случае успеха (автоматически):  Web.room=x | "Успешно! Ваш номер комнаты x"  "Нет места" |
| create(n, x) | Вместимость комнаты и номер комнаты | "Такая комната уже существует"  "Успешно!" |
| join(k, x, names, parms) | Количество игроков с данного устройства, номер комнаты, набор имён, ключ генерации карты | Присоединение к комнате, в случае успеха (автоматически):  Web.room=x  Web.index=index lab.names[x]+=names  lab.parms[x]=parms | "index"  $"Эта комната столько игроков не вместит. Свободно {n - index} мест"  "Такой комнаты нет" |
| далее: x=Web.room | | | |
| rwait() |  | Ожидание заполнения комнаты x | "0" или "1" |
| names() |  | Вернуть lab.names[x] | lab.names[x] |
| parms() |  | Вернуть lab.parms[x] | lab.parms[x] |
| delete() |  | Удалить комнату x | "0" |
| delete(x) | Номер комнаты |
| clear() |  | Очистить комнату x |
| clear(x) | Номер комнаты |
| set(str) | Строка | lab.action[x]=str | "0" или "1" |
| get() |  | Вернуть lab.action[x] | lab.action[x]  или "1" |
| wait() |  | Ожидание, пока все обработают ход i-ого | "0" или "1" |

|  |  |
| --- | --- |
| Переменные класса Web | Назначение |
| res | Результат действия функции; гарантируется, что становится пустой до запуска потока (т.е. необязательно самостоятельно выполнять Web.res="" перед каждым использованием функции) |
| room | Номер комнаты (int от 0 до 99) |
| index | Индекс игрока в комнате (int от 0 до n-1) |

bool online ... /\*псевдокод-идея, как можно попробовать реализовывать игру\*/

online = true ...

/\*сейчас ход i-ого игрока\*/

if (index<=i && i<index+k) //k - число игроков на данном устройстве; если online == false, то index=0, k=players

{

    /\*стандартный игровой процесс\*/

    step =...

    side =...

    /\*доходим до получения результата его хода\*/

    gameans =...

    if (online)

    {

        string str = step + delimeter + side;

        Web.set(str);

        bool cycle = true;

        while (cycle)

        {

            /\*меняем кадры\*/

            if (Web.res == "");

            else if (Web.res == "-1")

            {

                /\*проверьте подключение к интернету\*/

                Web.set(str);

            }

            else break;

        }

    } ...

}

else /\*заметим, что в эту часть кода можно попасть, только при online==true\*/

{

    /\*скроем все клавиши выбора хода от пользователя\*/

    Web.get();

    bool cycle = true;

    while (cycle)

    {

        /\*меняем кадры\*/

        if (Web.res == "");

        else if (Web.res == "-1")

        {

            /\*проверьте подключение к интернету\*/

            Web.get();

        }

        else if (Web.res == "1") Web.get();

        else break;

    }

    Parse(str, out step, out side);

    /\*обрабатываем step, side как будто их пользователь и ввёл\*/

...

} ...