Project

|  |  |  |
| --- | --- | --- |
| **Classes** | **Methods** | **Description** |
| **User** | isAuthed() | Checks whether user is authed or not |
|  | isAdmin() | Checks whether user is admin or not |
|  | getBaseID() | Returns user’s base ID |
|  | register() | Makes new user |
|  | login() | Logging user in |
|  | logout() | Logging user out |
|  | setRole() | Sets user’s role *(admin, user, other)* |
|  | getRole() | Sets user’s role. *Alternate to isAdmin() if roles more than 2* |
| **Base** | structureList() | Returns list of builded structures |
|  | isActive() | Returns whether is base produce or been under attack |
|  | upgrade() | Upgrade base at one level up |
|  | produce() |  |
|  | getOwner() | Returns ID of base’s owner |
|  | makeStructure() |  |
|  | setTask() | Sets task for base |
|  | getTask() | Returns current task of base |
| **Building** | upgrade() | Upgrade building at one level up |
|  | setTask() | Sets task for building |
|  | getTask() | Returns current task of building |
|  | getType() | Returns type of building |
| **Squad** | getOwner() | Returns ID of squad’s owner |
|  | getUnits() | Returns list of units in that squad |
| **Kernel** | tick() | Call every second. *Alias of routine() function.* |
|  | setGlobalVar() | Sets global variables such as starting resources count |
|  | getGlobalVar() | Sets global variables |
|  | makeStats() | Making summary based of statistics of all players |
|  |  |  |
|  |  |  |
|  |  |  |

**Database tables:**

**User**: id, nickname, password, role, wins, loses

**Base**: id, name,owner, [resources], [units], coordX, coordY, level, isActive

**Building**: id, type, level, baseid, currentTask, finishTime

**Squad**: id, owner, moveFrom, moveTo, startTime, finishTime, [units], [shipments]

**Routine loop**:

* Executing every second (*mb 2 or 3 seconds due to optimizing*)
* Adding resources to every base
* Looking for battles
* Proceed battles
* Reassign tasks
* Producing units

Game Setting

**Structures**:

|  |  |
| --- | --- |
| Energy station | Making energy as resource |
| \*name\* | Making credits as resource |
| \*name\* | Making units |
| \*name\* | Sets current population limits of base |

**Units**:

|  |  |
| --- | --- |
| Drone | Weak unit |
| Jet | Normal unit |
| Warship | Strong unit |

**Resources**:

* Credits
* Energy
* *Population limit of base*

**Base’s level**:

* Raising defence power of base
* Raising resources limits of base

**Level-based formula of price** *(starts with level 2):*

Price at current level = (base price) \* 1.05 + (0.96)in power of level

UI

Here goes some UI sketches:

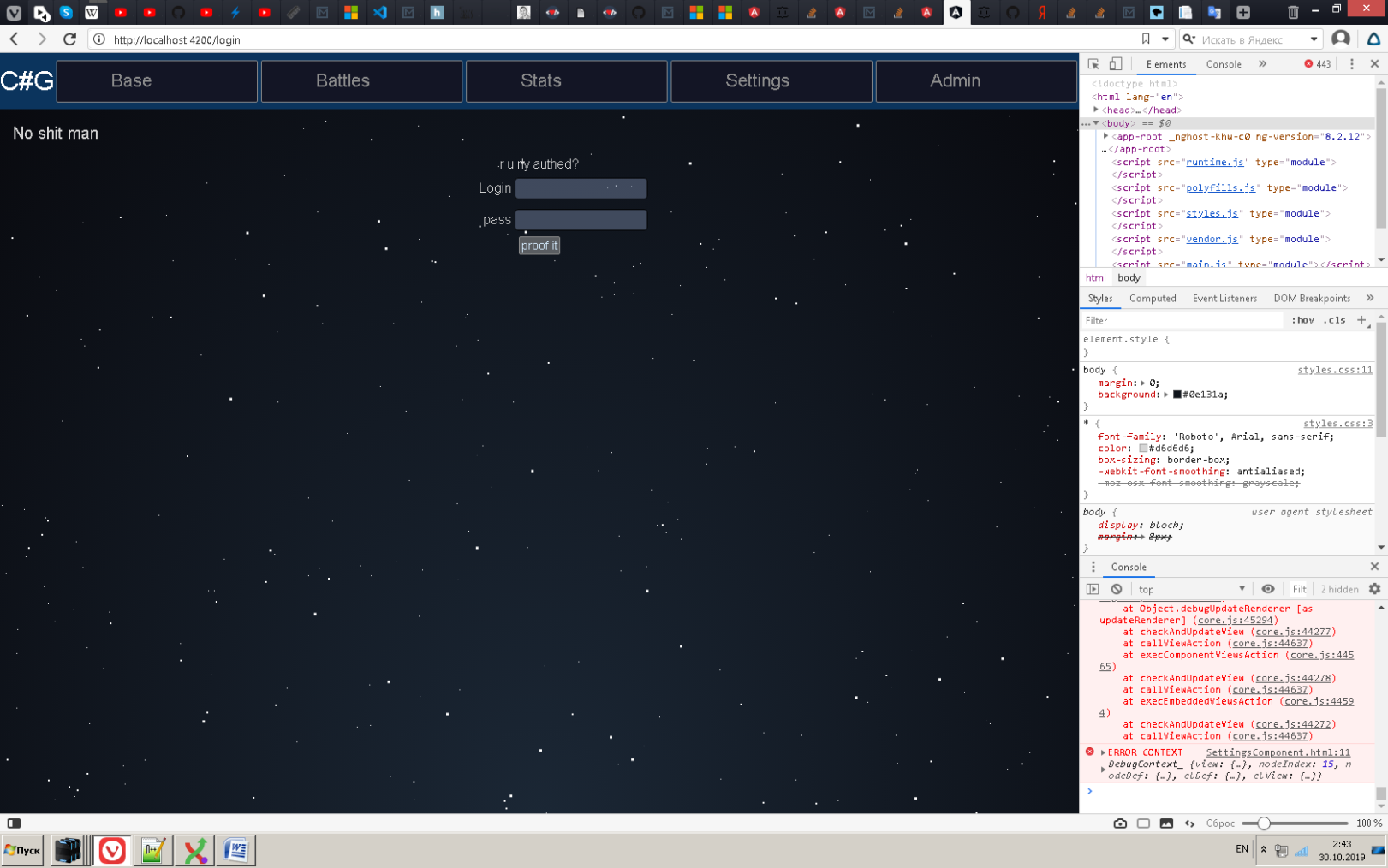
BASE SECTION:

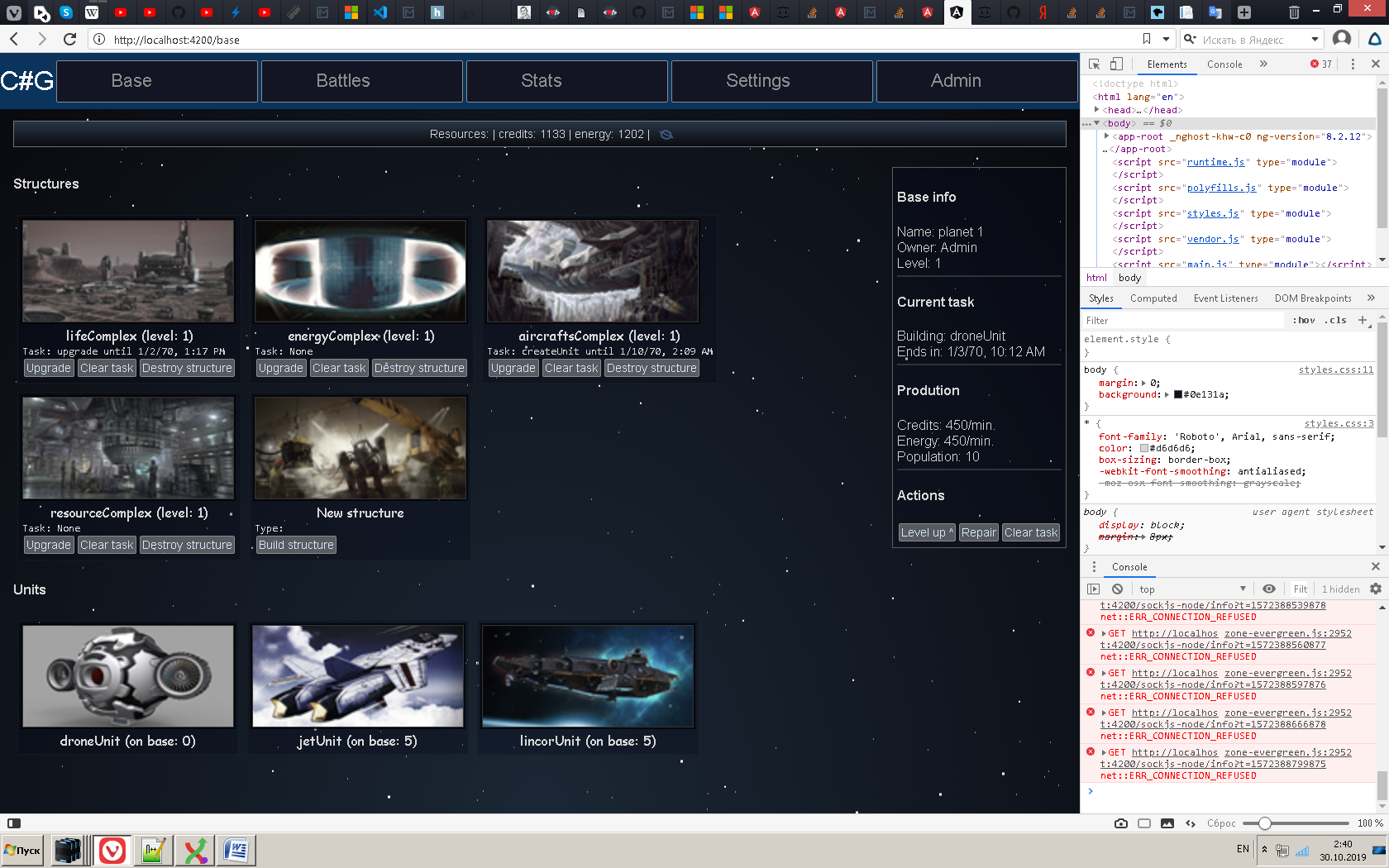
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Base** | Battles | Stats | Settings | Admin | | Resources [credits: 000| energy: 000] | | | | |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Structures   |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  | | --- | | *(Some pic here)* | | Name of scruct  Level: 1 | | |  | | --- | | *(Some pic here)* | | Name of scruct  Level: 1 | | |  | | --- | | + | | Add new struct | | | Owner: NNN  Level: 000  Prodution  Credits: 000  Energy: 000  Population: 000 | |

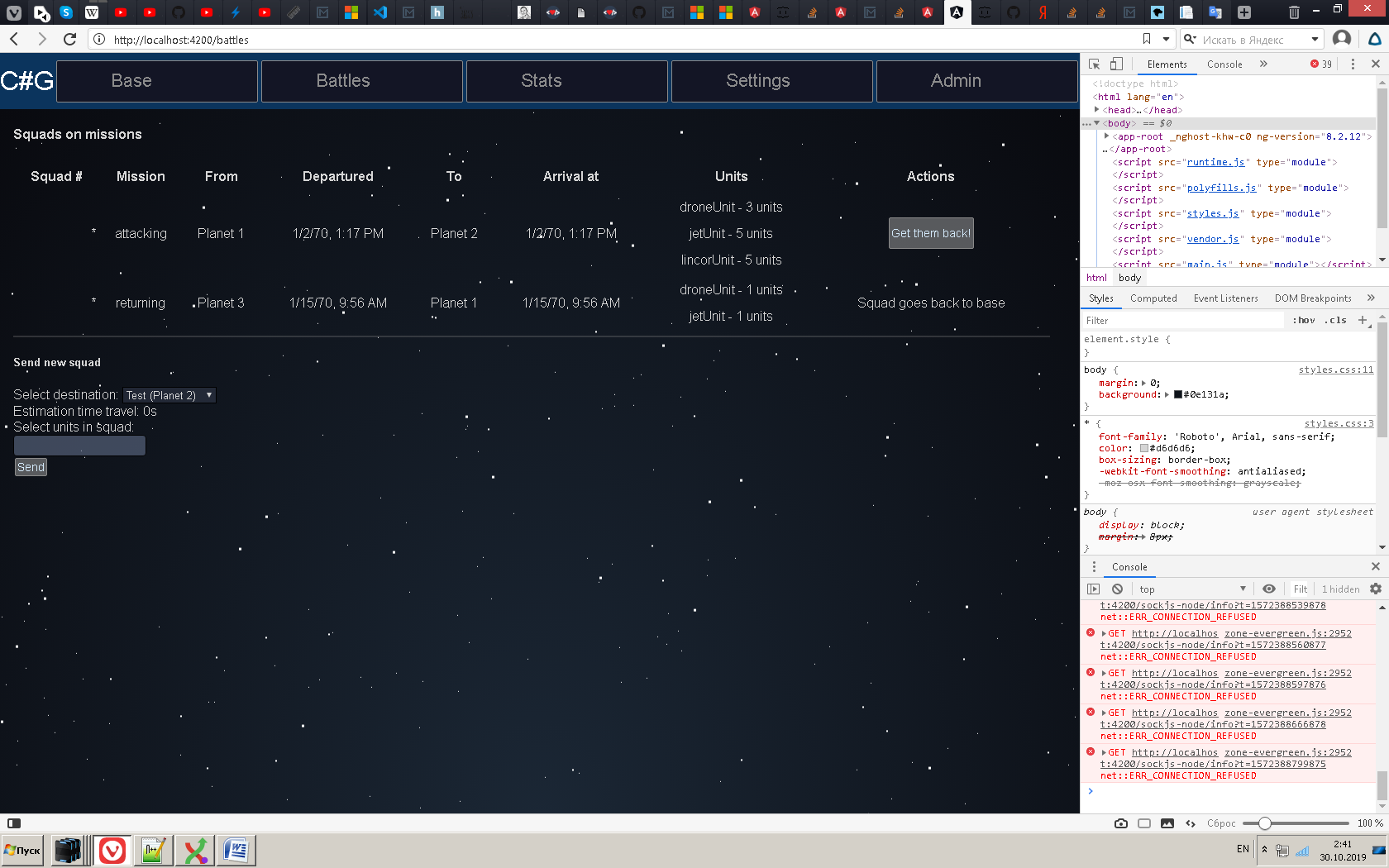
STATS SECTION:

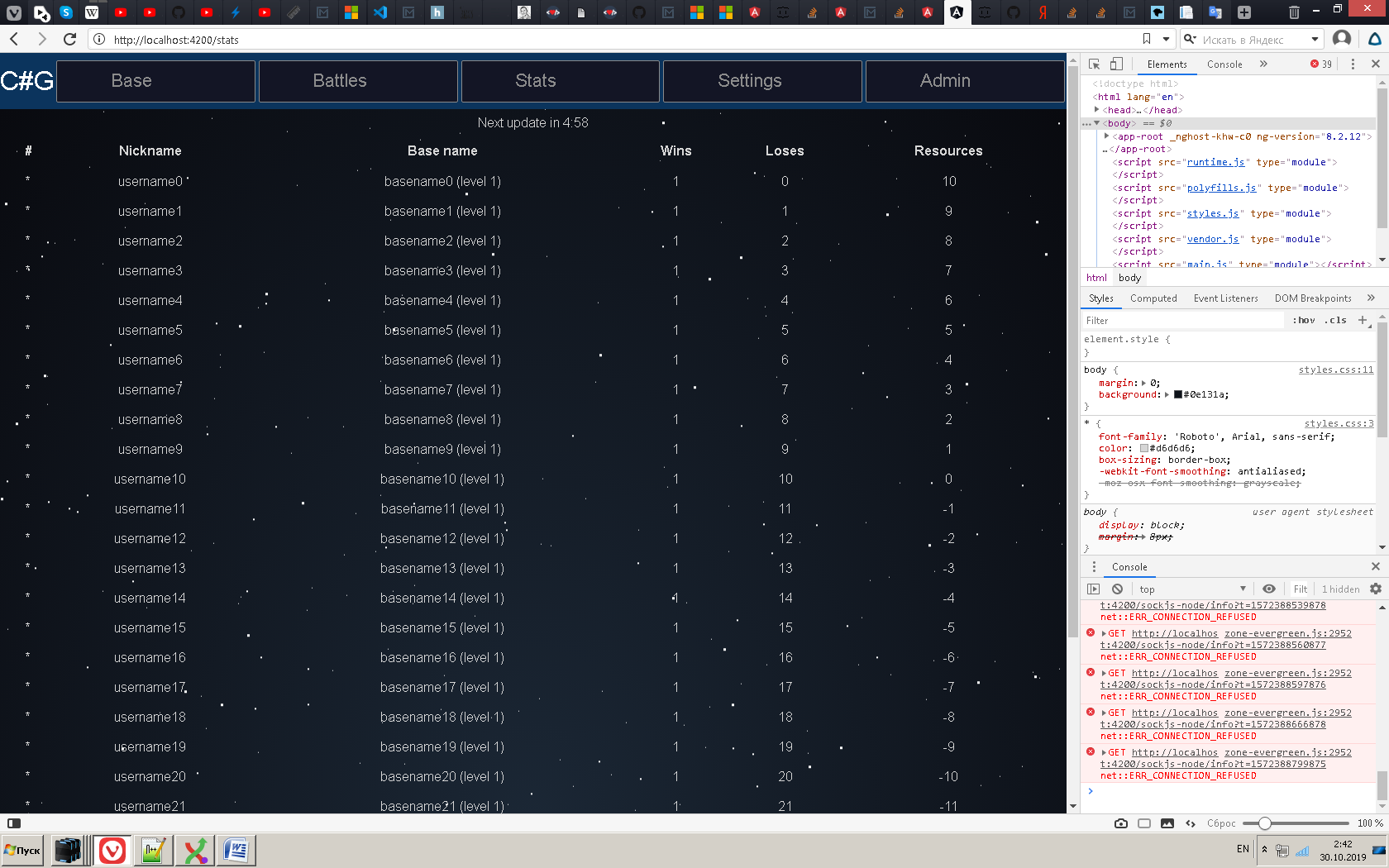
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | Base | Battles | **Stats** | Settings | Admin |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Stats   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Place | Nickname | Base | Wins | Loses | Resources | | 1 | Qwe | Base 1 (level 2) | 11 | 0 | Credits: 000  Energy: 000 | | 2 | Wdas | Base 2 (level 1) | 1 | 0 | Credits: 000  Energy: 000 | | 3 | Asd | Base 3 (level 1) | 1 | 0 | Credits: 000  Energy: 000 | | 4 | Asde | Base 4 (level 1) | 1 | 1 | Credits: 000  Energy: 000 | | 5 | S | Base 5 (level 1) | 1 | 1 | Credits: 000  Energy: 000 | | 6 | Sdfe | Base 6 (level 1) | 0 | 13 | Credits: 000  Energy: 000 | | |

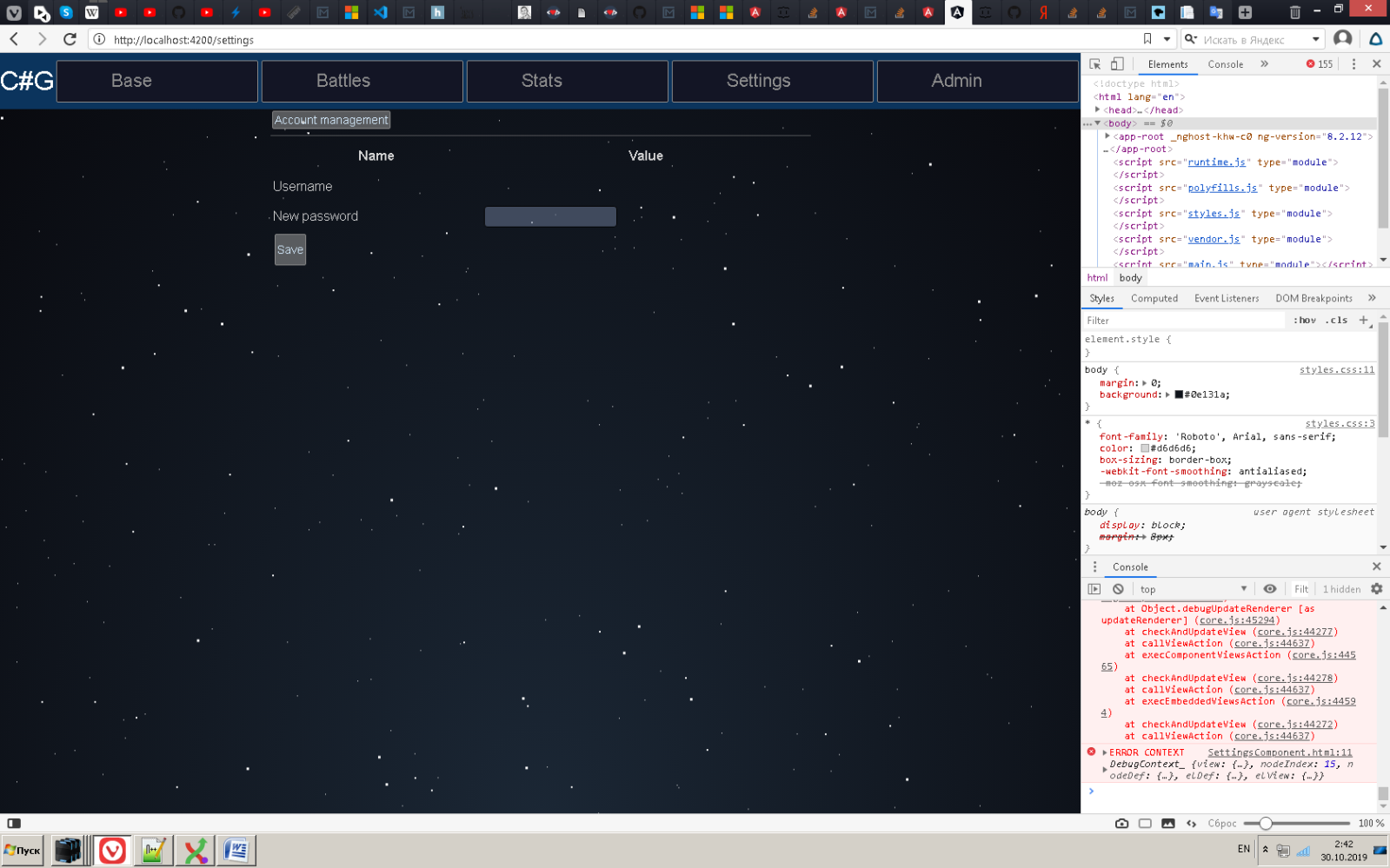
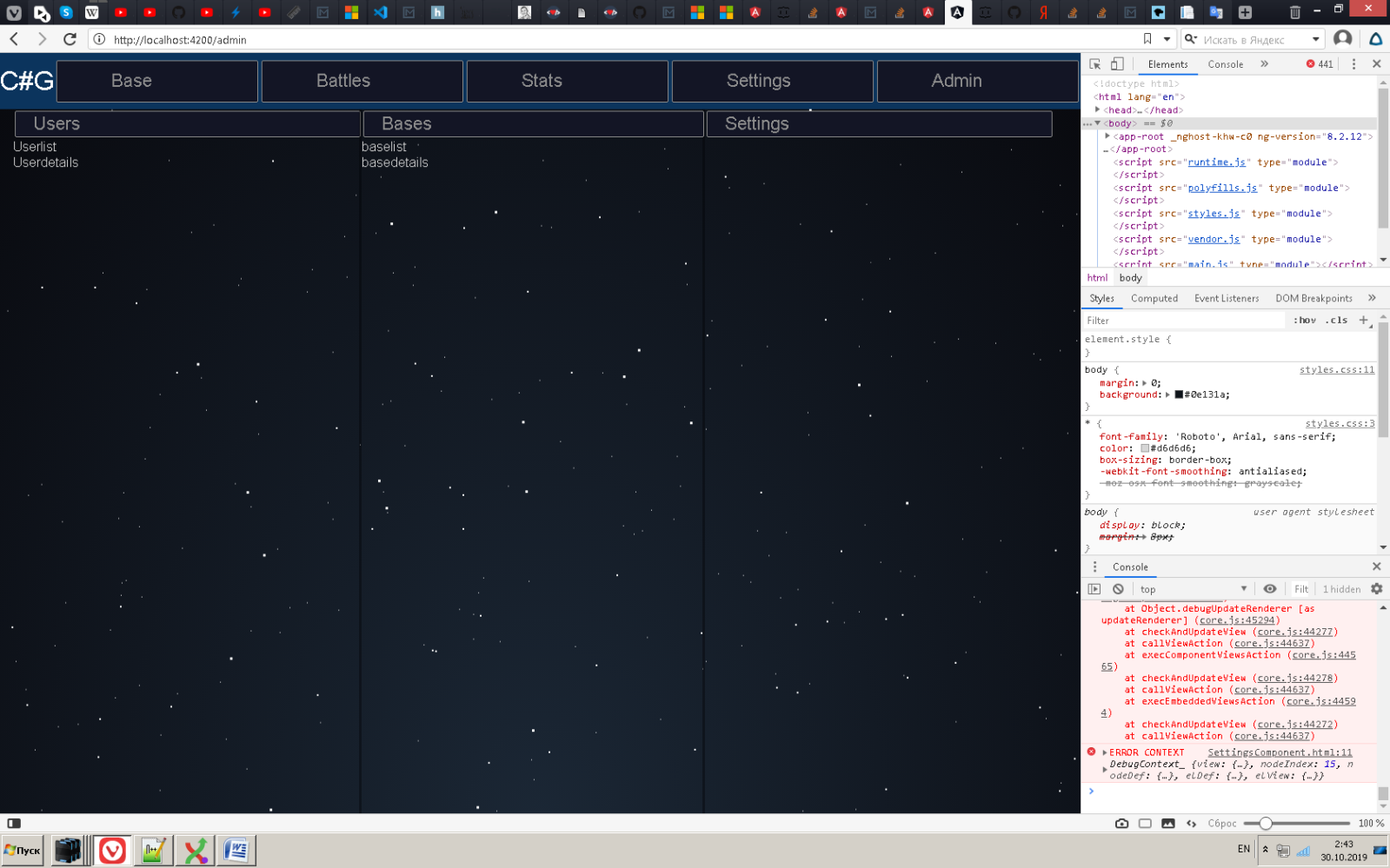
And here is real in-dev UI

LOGIN SECTION: 

BASE SECTION:

BATTLES SECTION:

STATS SECTION:

SETTINGS SECTION: ADMIN SECTION:

Tasks and schedule

|  |  |  |  |
| --- | --- | --- | --- |
| **Tasks** | **Goal** | **Deadline** | **Done** |
| UI | Make simple UI at Angular2+ | 22.10 | 22.10 |
| WebApi | Make basic methods to working with UI back-end | 27.10 | 29.10 |
| WCF | Make connection from WCF to WebApi | 30.10 | 05.10 |
| Business-logic | Build the main logic of game as DLL file | 07.11 | 05.10 |
| DAL | Connecting to DB | 13.11 | 05.10 |
| Finishing | Makes project better, fixing bugs | 19.11 | 19.11 |
| Show up whole the project |  | 20.11 | 20.11 |