

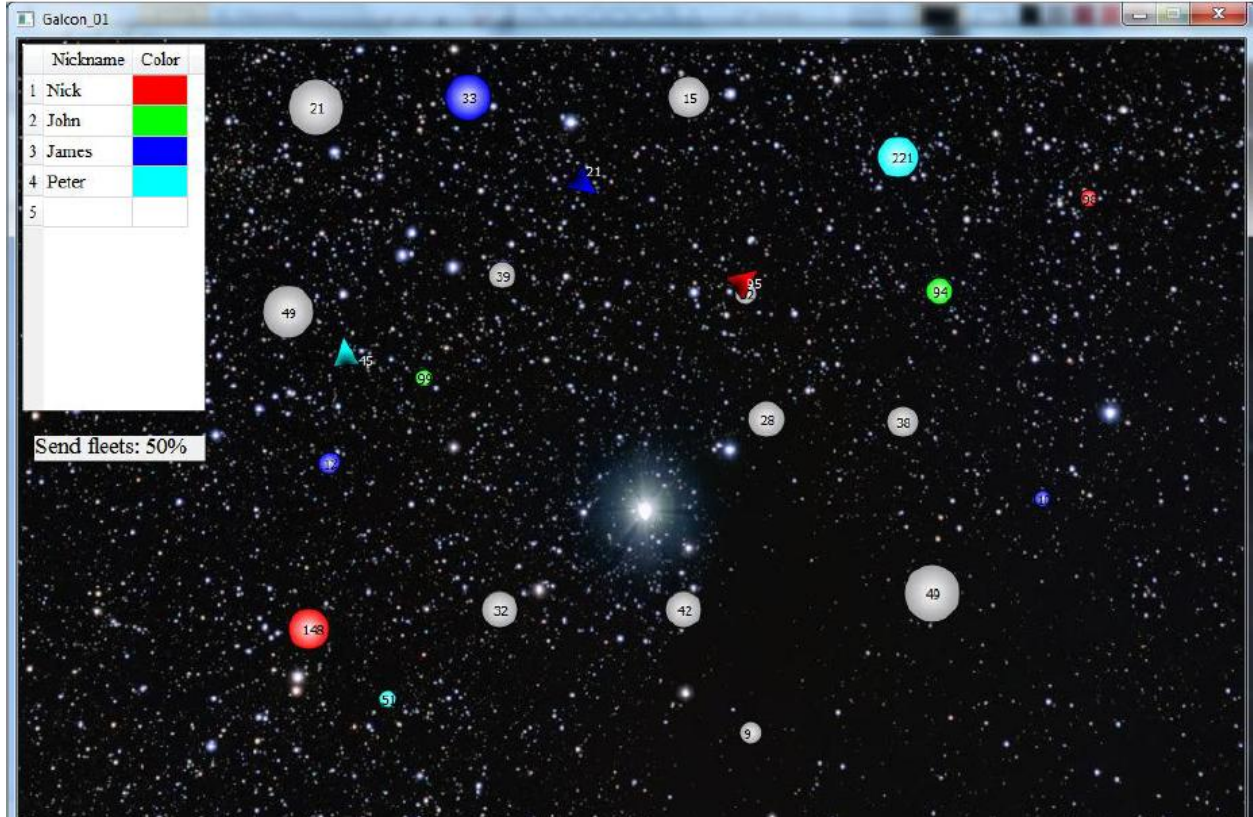
Assignment: This is a group project assignment. The goal is to create multiplayer online game Galcon (see the original game here: <http://www.galcon.com/flash/>) using Qt. You need to write a client side application (server side application is available for downloading). It must support all the key requirements:

- 2 or more players;
- Each player will has one or more starting planets;
- Players send units from their planets to other planets in a attempt to capture them;
- All captured planets automatically generate new units, with larger planets having a faster growth rate than smaller planets;
- Player that eliminates all other players from the map – wins.

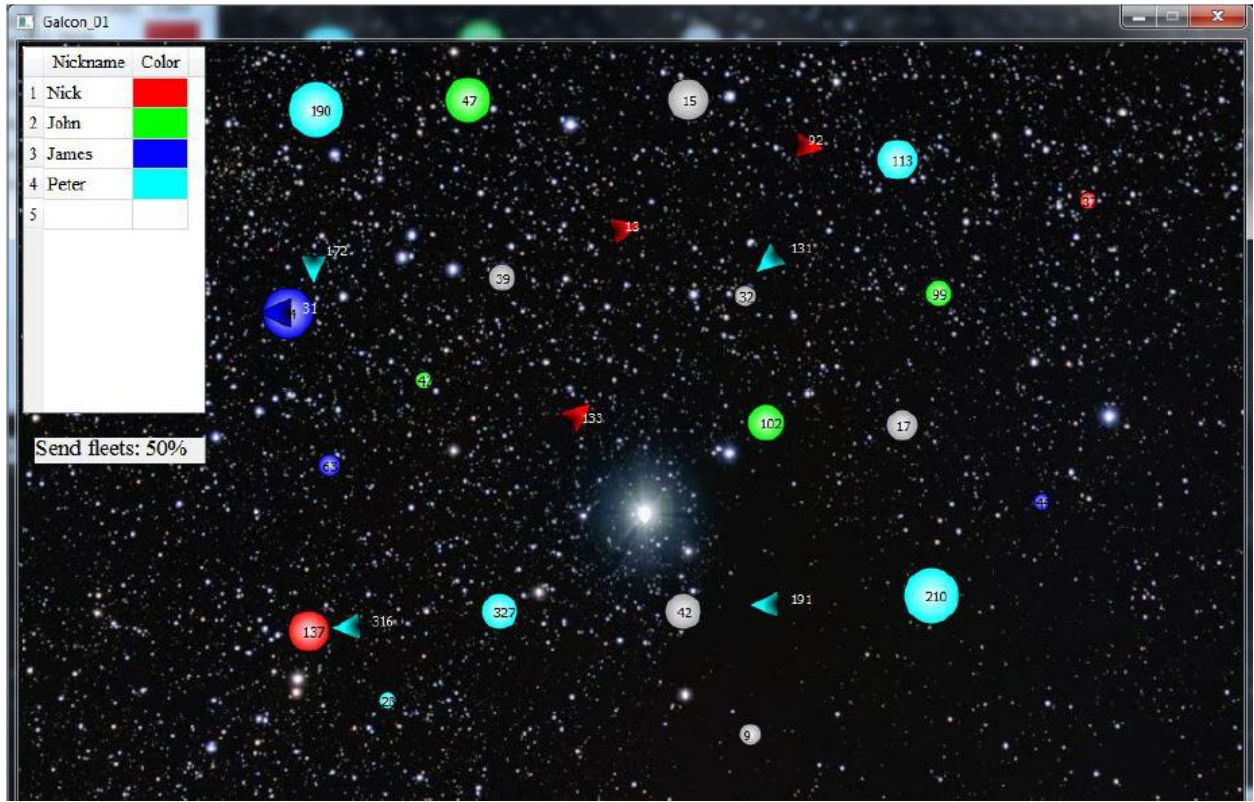
Solution:

Here are the screenshots of a game with 4 participants (PC platform):

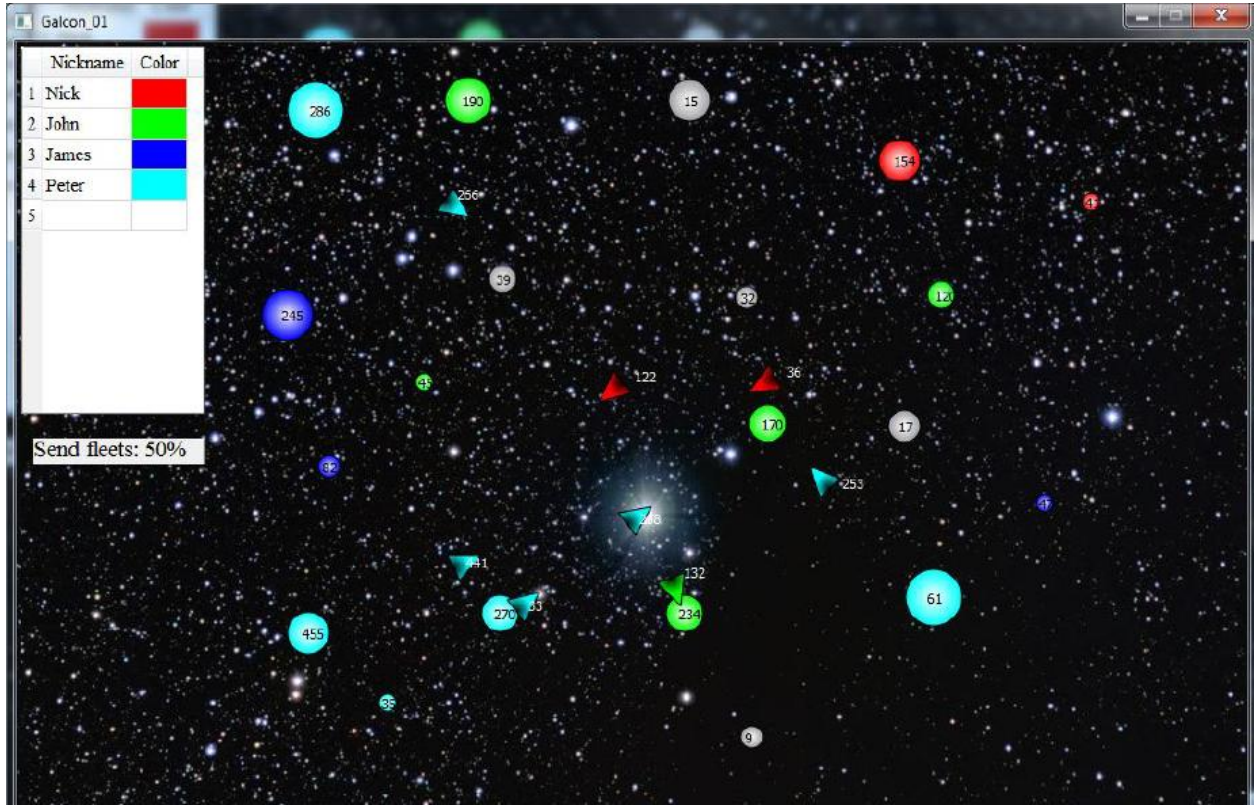
Screenshot 1:



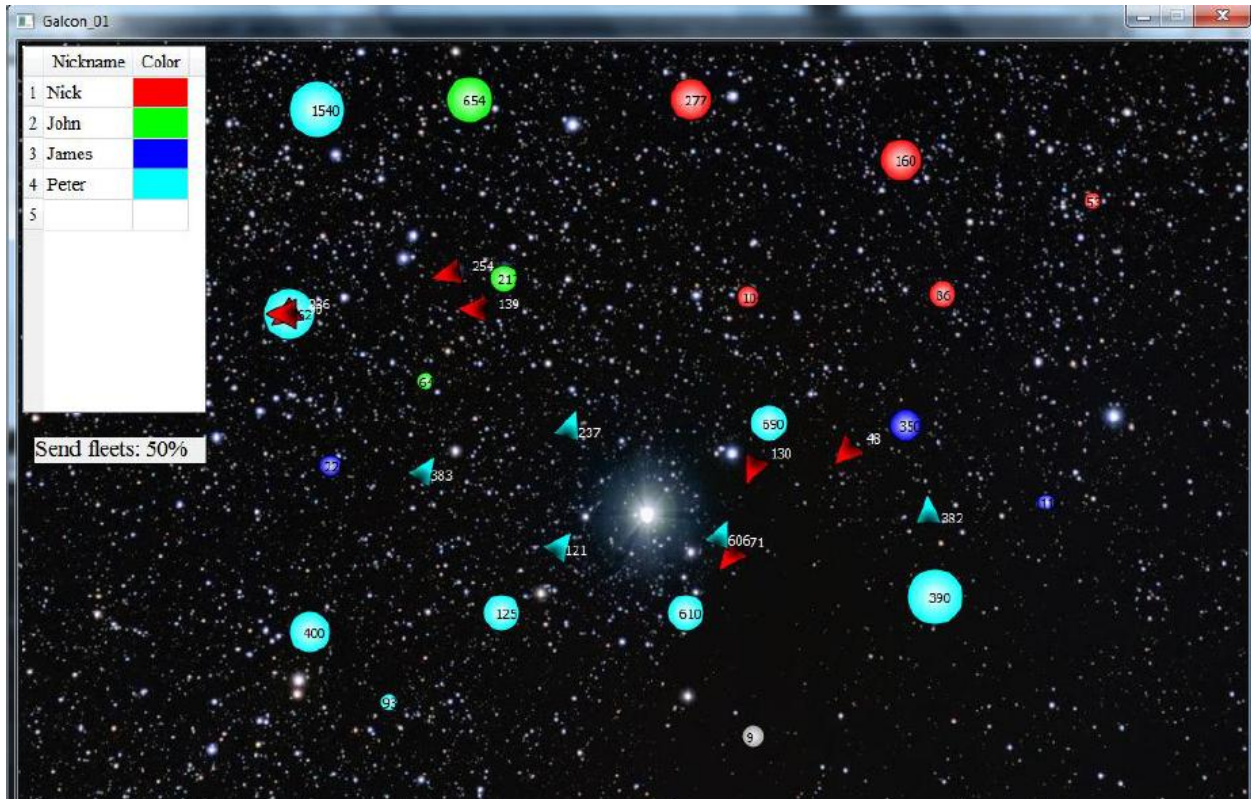
Screenshot 2:



Screenshot 3:



Screenshot 4:



Below is a part of the **source code** from the Galcon project:

Map.h:

```
#ifndef MAP_H
#define MAP_H

#include <QVector>
#include "Player.h"
#include "Fleet.h"
#include "Planet.h"
#include <QSize>
#include <QPoint>
#include <math.h>
```

Assignment 4 Student

The way to your success!

```
#include "message.h"  
#include <QTime>  
#include <QMap>  
#include <QDebug>
```

```
using namespace MessageDecl;
```

```
class CMap
```

```
{
```

```
private:
```

```
    QSize m_size;
```

```
    QMap<int,CPlanet *> m_planets;
```

```
    int m_moveSpeed;
```

```
    int m_increaseCoef;
```

```
    QMap<int,CPlayer *> m_players;
```

```
    QMap<int,CFleet *> m_fleets;
```

```
    int MyId;
```

```
    QTime m_time;
```

```
public:
```

```
    CMap(){}
```

```
    CMap(QSize size, int IncCoef, int MovSpeed, QMap<int,CPlanet *>  
planets,QMap<int,CPlayer *> players);
```

```
    void Init(CInitMessage *);
```

```
    void ProcessStateMessage(CStateMessage *);
```

```
    void ProcessMessage(CMessage *);
```

```
    void ProcessUserIdMessage(CIntMessage *);
```

```
    void IncreasePlanetShips(int);
```

```
    void MoveFleets();
```

```
    void CreateFleets(){}
```

```
    int GetNumPlanets()
```

```
    {
```

```
        return m_planets.size();
```

```
    }
```

Assignment 4 Student

The way to your success!

```
CPlanet* GetPlanetById (int id);
 QMap<int,CPlanet*>::iterator GetPlanetBegin()
 {
     return m_planets.begin();
 }

 QMap<int,CPlayer*>::iterator GetPlayerBegin()
 {
     return m_players.begin();
 }

 QMap<int,CFleet*>::iterator GetFleetBegin()
 {
     return m_fleets.begin();
 }

 QSize GetSize()
 {
     return m_size;
 }
 int GetNumFleets(){return m_fleets.size();}
 void GetFleetById(int id,QPoint & pos, QPoint &DestPos, int & OwnerId,int
 &NumShips);
 CFleet * GetPointerFleetById(int Id);
 void Refresh()
 {
     IncreasePlanetShips(m_increaseCoef);
     MoveFleets();
 }
 void DeleteFleet(QMap<int, CFleet *>::iterator iter){delete iter.value();
 m_fleets.erase(iter);}
 double DistanceBetweenPlanets(int Id1,int Id2);
 int GetNumPlayers()
```

Assignment 4 Student

The way to your success!

```
{  
    return m_players.size();  
}  
int GetMyId()  
{  
    return MyId;  
}  
QString GetPlayerNameById(int ID);  
};  
#endif // MAP_H
```