SettleIn is the rewrite of Sea3D in .NET 3.5. The application is designed with a visual representation of the reality in mind: all items a player would expect on a real life board should be visualized the same way.

SettleIn consists of a few components. First, we have a game engine for the original game, along with the SeaFarers extension. The second component is MapCreator. This is an interface for players to design their own boards. Third, the server component will be responsible for hosting the games.

Key classes which are used in all three components are Board and the Hex class hierarchy. The Board represents the board as displayed to the user. Board has an internal instance of Hexes, which is a data structure to keep track of all hexes. As can be seen in the class diagram Hexes.cd, Hex is the base class for all Hex types. One special hex type, NoneHex, is a representation of a hex that shouldn’t be displayed in the board game. Due to the nature of how Hexes is implemented, this type is necessary to create non-square board layouts.

Another key class in the shared classes of SettleIn is Piece. Although yet only a handful classes uses this as a base class, this class should be used in the future development as a base class for DevStack, Bank, Hand, etc.

The user interaction with a board takes place in the base class 3DBoard, which derives from ViewPort3D. This class holds all shared logic for the visual representations of a game board, a map editor board or a read only board for viewing purposes. Derivers from 3DBoard are 3dMapEditor, S3DGameBoard and BoardViewer. These classes implement logic to their specific purpose. For example, the functionality of placing a port is kept in 3dBoard, since both S3DMapEditor and S3dGameBoard would both use this functionality.

Map Editor

The mapeditor is a visual representation of real life in where the user would be designing a new board layout. Since the user is given a lot of freedom in creating this layout, before saving a designed board, the board is checked against a set of rules. The IRule interface specifies the signature of these rules. A simple example is KnownName: this class implements IRule, and checks if the user didn’t put an official name on the board, like “New Shores”.

GameBoard

Gameboard is a visual representation of the board as one would see it in real life. The gameboard consists of the actual playing board, the development card stack, the bank, the dice, the chit bag for hidden hexes, the hex stack for hidden hexes, the robber and pirate, stack of bonus victory points, the stock items like ships, roads, towns and cities and what I forgot.