Teamname: THE Settlers of Catan

Github: https://github.com/NumairGoldeye/cs194-project

Charlie: charlesy@stanford.edu (username: Charlie-y)
Chase: chasebasich@gmail.com (username: chasebasich)
Chris: chrisvh1@stanford.edu (username: chrisvh)
Khalil: flamewingman28@gmail.com (username: NumairGoldeye)
Kevin: dingruoyu@gmail.com (username: ruoyud, sunet: ruoyud)

Chat Email:
Khalil: khalilgriffin@gmail.com
Chris: chrisvh1@stanford.edu
Charlie: Charlieeyang@gmail.com
Chase: cdbasich@stanford.edu
Kevin Ruoyu Ding: dingruoyu@gmail.com sunet:ruoyud

http://www.when2meet.com/?2472671-MKm1h

Description of the Project:

Our project will be an online, multiplayer *The Settlers of Catan* interface. This interface will allow people to play with each other remotely or an AI. The most important feature is an ability to follow the rules logic. Users will obviously need to be able to play a game from start-to-finish, following the rules of *The Settlers of Catan*. However, to make our system stand out, we will have several other features, including:

- **A beautiful graphical interface:** The system will use 3D graphics in order to create an immersive experience for the user. The board tiles, resources, robbers, and all game pieces will have 3D graphics that keep the user engaged.
- **Intuitive and convenient user controls**: Our system will have user controls that make the game easy to play. Whenever it is the user's turn, the system will clearly notify him or her. It will provide a clean way to manipulate resources, build properties, and interact with other players. This will allow for seamless gameplay without frustration; something that the physical board game can struggle with. We may also enable notifications, such as through

email or text, for significant events, like when a spot opens up in a game, or when it becomes the user's turn.

- **Customizable games**: There are several different options that can be used for games. For example, while almost all in-person games are played in real-time. However, since the Settlers of Catan can take hours, and we recognize not everyone has that much time, we will also allow long gaps between moves. One option used will be the ability for whoever creates a game to specify a maximum gap between moves. Real-time games might use a few minutes, whereas long-term games could use hours or even days.
- **A user profile system**: Each user will have a profile the includes information like their number of wins and losses, achievements, and information about his or her previous games.

Need for the Product:

The Settlers of Catan is a wonderful, well-balanced, and critically acclaimed board game guaranteed to entertain you and up to three of your friends, as long as you own the game and are in the same room as each other. However, the game is not easily accessible. By creating an faithful online reproduction of the game, we will allow people to play this game when they are apart and without the physical board. A computerized version of *Settlers of Catan* also allows players to quickly setup and organize the pieces and board to play on, a usually lengthy process.

Discussion of Competing Products:

The main competition is the *The Settlers of Catan* board game which our product is based on. The obvious draw of our product is that it is on the computer and it is networked. This allows people to easily play with friends or with strangers anywhere in the world at any time. The clean interface will also automate many of the mundane tasks in the game, to help provide a faster flow of the game than would be possible playing with the board version of the game.

There are already electronic versions of the game available. However, these versions simply reproduce the 2D board online, whereas our version will use 3D graphics for a more visually appealing game. Furthermore, our version of the game will have options for both real-time gameplay and asynchronous gameplay to accommodate users who do not have blocks of time to devote to playing, but can play over days or weeks.

Ultimately, gameplay and the rules behind the game will remain largely unchanged, so the main draws to our version of the game will be aesthetics and convenience.

Potential Audience:

The target audience are people who enjoy playing the *The Settlers of Catan* board game and want to play with either strangers on the internet or friends who are unable to play in person. With both real-time and asynchronous gameplay, our product caters to players with varying schedules and opportunities to play a game. A great example of a potential audience is a group of friends from school who are on break and wish to stay in communication and play their favorite game together.

The target audience has the potential to be quite large. There is a small community of about 1500 people that play Settler's of Catan online through playcatan.com, but there is a vast community of people who enjoy playing the board game, in particular students at universities. The level of technical sophistication of the target audience varies, but is quite irrelevant to the design of our product; it will cater to people of all technical backgrounds.

Major Technologies Used:

- **SQLite on Mono Server:** We will be using a Mono backend to support a database as well as allow for communication between Unity and the backend. Since Unity employs Mono, using Mono as the framework for the webserver will simplify the calls to the database. Mono supports many database platforms, including SQLite3, which we will use.
- **Unity:** We will be creating the game using Unity as our engine. While Unity does not render as well as other available engines (for example, Unreal), it is easy to use and has good built-in browser plugin for easy networking. Some of our members already have familiarity with Unity and it has a lower learning curve than other engines.
- **Online Asset Stores / 3DSMax:** We will ideally find free 3D models online. When this fails we will either purchase models or create our own using Autodesk 3ds max.
- **AWS:** We will use AWS to host our game
- **Unity Networking:** Unity has built in networking tools. Although other networks such as Node.js can be used with Unity, we foresee the built-in tools being sufficient for our needs.

- **C#:** Unity has 3 options for programming language: 1) C#, 2) Javascript, 3) A combination of the two. We will be using C# because the group overall has the most experience with systems languages, and we don't want to use a combination for the game itself because it would introduce unnecessary complexity.
- **HTML/CSS/Javascript:** We will use these for developing a front-end to the website that will be hosting the game. This is where players can register, find games, and check statistics.
- **Github:** We will use Github for our version control, per the course requirements.

Resource Requirements:

Although we can host on Heroku if need be, we would appreciate AWS credits for a more powerful server. This would allow for our product to support more games and more players at a given time than possible with Heroku.

We might need to purchase 3D models if we cannot find free models as we will not have time to create the more complex 3D models.

Potential Approaches:

We will use a combination of web technologies to build the infrastructure in the form of a web app.

Before we start, we will identify critical technologies used in building settlers of Catan and acquaint ourselves with those frameworks with a set of learning materials. After we are comfortable setting up our game framework using those technologies, we will start building the infrastructure of the game and do version control on github.

First, we are going to focus on the game mechanics. The settlers of Catan involve a set of nuanced game rules that involve multiple elements of the game. ie, how to draw a card? how to build a road? how to set up or upgrade a building? This will involve building basic graphics, the backend database structure for storage and the turn-based mechanics.

After the turn by turn game mode by 1 end point navigation is finished, we will focus on the networking aspect of the game such that two players from two computers can play together over the web.

Then we will develop or acquire professional graphics to be combined into the web app to build better visual experience. At the same time, we will be adding AI elements for a computer mode, where users can try to beat the computer implemented by our algorithms.

Towards the last 3 weeks of the development, we will start test runs with real Catan users and iterate based on user feedbacks. These will also involve guidance features where new users can follow the instructions to learn how to play the game easier.

In terms of our approach to development, we will use an agile structure. We focus on face to face communication and every week we will meet in person and code together. During that weekly meeting we will review our progress and also reassign the tasks to do every so often and ensure working software at every milestone.

Assessment of Risks: Kevin (with a little help from everyone)

- The AI might not be competitive and we might have very limited time implementing the computer mode. We're currently unsure how strong an AI for this game can be, and what methods would be best to use, so we run the risk of having a weak AI.
- Game development is hard.
- Graphics can be hard. There is limited public models related to settlers of Catan available, so we might need to build our own models on top of existing resources.
- Team coding is hard. We need to modularize the entire project into different components.
- Network might not be able to handle multiplayer load.
- Github can be bloated with the large unity files, as well as difficult to synchronize a large Unity project that exceeds the allotted space on Github.
- Compilation issues in Unity: if we prefer different languages, there are some issues where unity compiles certain files before others, so we could get weird bugs with using both C# and Javascript.
- Version control issues in Unity: Considering large asset sizes and some internal file management that Unity does.

- Unity does not come with any standard for databasing and it can be difficult to setup a SQL database to work with Unity.

Next Steps:

The first step that we will take is to introduce the group to all the technologies, languages, and platforms we will be using. This will require some of our group members to become comfortable using tools such as Unity and new languages, such as C#.

Next we will try to define a basic outline for the entire project by fleshing out all the necessary components and the interfaces between the components that we'll require. After creating this framework we will use it to divide the work between the five of us.

We will set weekly milestones as we go, but the first milestone would be to get a set of prototypes on all the different systems and have them be able to communicate with each other, i.e: a web server that responds to url with a Unity web-player. From then on we will begin a series of iterations moving toward a final, polished products per the agile framework.