Complexity Research Initiative for Systemic InstabilitieS

FP7-ICT-2011-7-288501-CRISIS

Deliverable D5.3
Integration Plan

Author(s)	Tamas Mahr (AITIA)
Abstract	This deliverable, D5.3, documents the first release of the on-line game. The basic concept of the game was described in D5.1 Game architecture. In D5.1, we outlined a game that is backed by an integrated macro and financial agent-based simulation. The simulation was envisioned to contain bank, firm and household agents that interact with each other to form an economy. Players of the game control banks. They can decide to offer loans to firms or to other banks, they can compete for deposits of households and firms, and they can trade the shares of firms on a stock market. During the game, the players taking the role of a bank interact with the simulation and other (human or artificial) agents via markets. In particular, players submit orders to markets (deposit, loan, and stock markets), which are matched to counter orders submitted by fellow players, or artificial agents.

Distribution level	Public	**Status**	Final	**Version**	01
Contractual delivery date	31/10/2012	**Actual delivery date**		29/11/2012	

The CRISIS project is funded by the European Commission under the Seventh Framework Programme (ICT)

The On-line Game

AITIA International, Inc.

Table of contents

1 Introduction

This deliverable, D5.3, documents the first release of the on-line game. The basic concept of the game was described in D5.1 Game architecture. In D5.1, we outlined a game that is backed by an integrated macro and financial agent-based simulation. The simulation was envisioned to contain bank, firm and household agents that interact with each other to form an economy. Players of the game control banks. They can decide to offer loans to firms or to other banks, they can compete for deposits of households and firms, and they can trade the shares of firms on a stock market. During the game, the players taking the role of a bank interact with the simulation and other (human or artificial) agents via markets. In particular, players submit orders to markets (deposit, loan, and stock markets), which are matched to counter orders submitted by fellow players, or artificial agents. In D5.1, the game was essentially described as a participatory simulation which can be accessed through a web interface.
D5.2 Integration plan detailed the software architecture implementing the game as designed in D5.1. The architecture, as depicted in Figure 1, consists of a web portal with a game database, possibly multiple game engines consisting of a game adaptor and an integrated simulation each. Either at browsing the portal or playing the game, the user interacts with the system using his/her browser. How the different components (the portal, the game engine, the adaptor, the simulation, and the browser) communicate with each other is described in D5.2 in detail. The current implementation follows the D5.2 specification, therefore in this document we do not repeat these technical details.

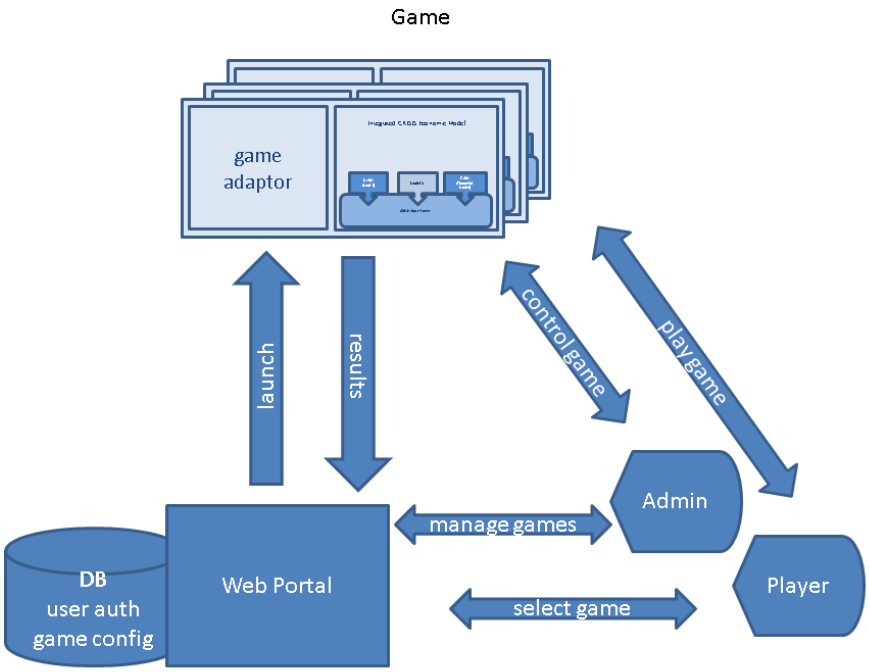

Figure 1 Game architecture.

This document describes the first release of the on-line game. This is the first playable version, meaning that the users can start and stop games, and can interact with the underlying

simulation. Despite that the basic functionality is there, the current version is more a participatory economic simulation than a game. The graphics is rather preliminary, and the underlying simulation has to be improved by several features to have a complete game. In this period, all effort went into realizing the base machinery (as described in D5.2) that allows browser clients to join an agent-based simulation and interact with it. Future releases of the game will improve the look as well as the gameplay.

The game portal is available at http://game.crisis-economics.eu. Here anybody can start the current version of the game, and access it either as admin or as normal player. The source code is separated into several different projects (game-common-config, game-adaptor, game-adaptor-api, game-engine, game-web), all of which is available in the mercurial repository of the project (https://www.crisis-economics.eu/scm/). To access the repositories, one has to provide a username and a password, the same username and password that is used for the project portal (http://www.crisis-economics.eu).

The following sections concentrate on documenting the current state of the integrated simulation the game is based on (in Section 2), as well as presenting the user interfaces that allow the players to interact with the simulation. First, Section 3 describes the portal pages the players visit to create or join games. Then, Section 4 introduces the game interface and the player controls as implemented in the current version. Finally in Section 5, a summary about the current state of the game is given.

2 Underlying model

As described in D5.1, the underlying model of the game is an integrated macro and financial simulation. This integrated simulation model is developed in WP8 in a collaboration of several partners. The model consists of Firm, Household and Bank agents, different kinds of contracts that connect these agents (e.g. bank account, or loans), and markets that constitute the primary way the agents make their contracts. In this section, we shortly summarize the above listed entities of the model, and discuss their role in the game.

Let us discuss the markets first, as they are central in the model (see the mid-left part of Figure 2). The markets in the integrated simulation model are *limit-order-book* (LOB) markets, meaning that the submitted orders contain a *size* and a *limit-price* parameter. It is possible to submit limit orders (limit price > 0), or market orders (limit price = 0) to these markets. Depending on the sign of the size parameter, the orders are taken as sell (size > 0) or buy orders (size < 0). The interpretation of these parameters is different for the different types of LOB markets, but the matching mechanisms (limit-order or market-order matching) that actually use them are the same. In the model used for the game, there are four different financial market modeled as LOB.

The *deposit market* is used by banks to attract deposits, and by firms and households to locate the bank that pays the highest interest rate on deposits. On this market, banks submit buy orders with a limit price denoting the maximum interest they would pay on deposits. Households and firms submit sell orders with a limit price denoting the minimum interest they want to receive on their deposit. As they submit such orders when they consider moving their money from one bank to another, their natural limit price is the interest they receive from their current banks.

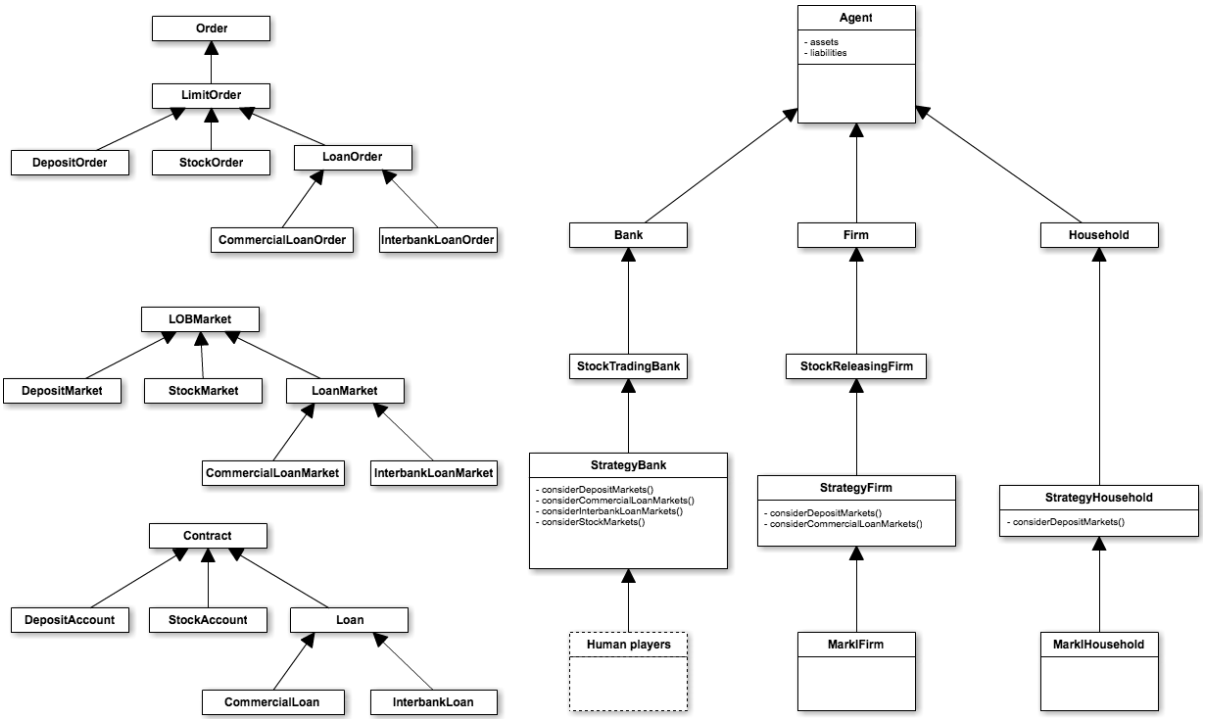

Figure 2 The integrated model of the game with the Mark I economy classes.

The *commercial-loan market* is used by firms to get loans to finance their production, and by banks to invest in the economy in the form of loans. Firms submit buy orders with a limit price denoting the maximum interest rate they are willing to pay for a loan. Banks submit sell orders with a minimum interest rate as limit price. Firms have to consider how much money they need (the size of the loan), and what price they can pay for it. Banks have to consider how much money they want to invest, and what the appropriate interest rate is that is lower than other banks' offer, but is high enough to make a profit.

The *interbank-loan market* is very similar to the commercial-loan market, the difference being that here banks submit both the buy and the sell orders. The interpretation of the size and price-limit parameters are the same, the decision on submitting orders to this market entails considering whether the bank needs more funds, or has excess money to invest. The answer to this question puts the bank on either the buyer or the seller side, with further considerations similar to those in case of commercial-loan markets.

The last market type in the integrated simulation model is the *stock market*. In the model, banks own shares of the firms, and they can trade these shares on the market. That is, on the stock market both sellers and buyers are banks, and the orders represent their intention to trade their shares in the given quantity for the given limit price.

After discussing the four different LOB markets present in the simulation model, let us now focus on the three different agent types (the right-hand side in Figure 2). These are the firm, the household and the bank agents. In the game, firm and household agents are artificial agents, while banks are controlled by human players. The economy of the game, including the firms and households, is based on the Mark I model developed in WP3. This model is improved by introducing the LOB markets as new means for the firms to acquire loans (on the commercial

loan market), and for households to put their money in the best bank (by using the deposit market). In addition to integrating these markets into the Mark I model, all other internal logic of the Mark I economy is preserved. Bank agents in the simulation model of the game are simple stubs, no real decision making logic is programmed into the bank agents, as the decisions are made by the players of the game.
As stated earlier, the integrated simulation models the economy and the financial sector as a unified network, where nodes are agents, and links are contracts. Firm, household and bank agents organize their contracts (their links to each other) as balance sheets, counting each contract either as an asset or a liability. To support this model, a couple of base classes have been implemented in the integrated model. Most notable is the *Agent* class which (in object oriented terms) is the ancestor of all three agent types. This base class provides the machinery to handle the lists of assets, liabilities and markets for the agent subclasses. The subclasses inherit this machinery, and specialize it to their needs. E.g. household agents store their bank accounts (which are a kind of contract) in their asset list, similarly to firms that store their loans in their liabilities list.
To summarize, while developing the integrated simulation model, a base library seems to emerge consisting of the basic functions that are required by all financial and macro models. These are the classes of the different contract types (deposit account, loans, stock accounts), the four markets and the base agent classes that can be used to derive more sophisticated versions.
After this review of the underlying model of the game, in the next sections we discuss the game portal and user interface of the game itself.

3 Game portal

The game portal is the central coordination point for CRISIS gamers and game administrators. Its two main functions are to facilitate game playing and to organise the CRISIS gamer community. In this section, we describe the already available features of the portal, as well as point out the directions of (near) future development.
Game administrators, once logged in, can configure previously uploaded simulation models in order to create different types of games. A configuration is the definition of model-parameter values that are used to initialize the simulation model at startup. Currently, game administrators can provide a configuration in the form of an XML pasted or typed into the corresponding field of the form. An example configuration is provided in Table 1.
Configured game types are listed on the main page of the portal allowing anybody to start a game. When a game is started, the portal provides two links to the game, an admin and a user link. The admin link can be used to access the game as an administrator which grants extra permissions to the player, such as starting and stopping the game. The user link should be used by ordinary players to play the game. This link is normally forwarded to people the game creator wishes to play with. Note, that the page displayed to the admin user might be totally different than that displayed to the ordinary players (e.g. some kind of a monitoring screen), or it can be rather similar featuring a few extra buttons for starting/stopping the game.

```
<?xml version="1.0" encoding="UTF-8"?>
<simulation xmlns="http://aitia.ai/game/simulation"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://aitia.ai/game/simulation simulation.xsd">
    <args>-seed 1</args>

    <parameter-values>
         <param name="numberOfBanks" value="10" type="int" />
         <param name="numberOfHouseholds" value="100" type="int" />
         <param name="numberOfFirms" value="50" type="int" />
         <param name="policyRate" value="0.02" type="double" />
    </parameter-values>

</simulation>
```

Table 1 An example model configuration.

In addition to the game types, the portal also lists the currently running games providing links to join these games. This is useful, for instance, if somebody needs to re-connect to a game due to a browser failure. Additionally, this can be used to collect players for a game, in a case when the user link has not been forwarded to players.

The above described pages are used to create games and to allow players to join the games. The other function of the portal is to organize the CRISIS gamers community. To this end, the portal features a structured user forum, where players can discuss game related topics, and game administrators can provide information about the games, and can answer questions about the games. The forum page consists of several forum categories. The 'Announcements' category contains posts from the game administrators related to the game portal as a whole, and to the release of new game types. Further categories, one per type of game, are provided to the community for general discussion.

The game portal in its current state is still very preliminary. First of all, it should receive a design that is aligned with the design of the game. Furthermore, some key functions like displaying the result of games, and displaying the scores of users should be added. Finally, the administrator interfaces could be further developed to make uploading and configuring the game models easier. After this introduction of the portal, in the next section we describe the GUI of the game the players see and interact with while playing the game.

4 Game UI

The game UI is displayed in a browser. Players, in principle, can use any browser to play the game. Effort has been put into making the game client run on the currently popular web browsers. We have tested the client in Chrome, Firefox, Opera and IE9 on Windows, and Chrome, Firefox, and Safari on Mac.

The game client is loaded into the browser when the user clicks on the link of the game, or types in the URL of the game in his/her browser. This can be sent to the players by e-mail, or they can find URLs to open games on the portal. The creator of the game can also use the admin link to connect to the game. The GUI of the admin is almost identical to the GUI of the

players, except that the admin user has some extra features like starting/stopping the game. The first screen the players are presented with is the *Lobby*. This screen is displayed as long as the admin starts the game. In the lobby, players wait for other players to join. Once the game is started, the main interface is displayed that has five pages: the *Home, Shares, Deposit, Loan, Interbank* pages. In the next sections, we discuss these pages in detail.

4.1 Lobby

As discussed above, the Lobby screen is displayed to a user after (s)he joined the game (clicked the link of the game). This screen shows the players already joined and the type of agents they control in the game. The players are currently displayed as numbers; this will change to user names in the future. The admin version of the Lobby screen (as presented in Figure 3) features a 'Start Simulation' button. This is used by the admin players to actually start the game. Normal players do not have this button.

Figure 3 The Lobby.

A possible future improvement of this screen is a chat that would allow the players to communicate while waiting. The chat functionality is usual in such game lobbies. It helps players to kill their time while waiting, and it also allows them to coordinate when the admin should press the start button.

After the game is started the main screen is displayed. This screen has five sub-screens, *Home, Shares, Deposit, Loan,* and *Interbank.* Each of these screens, except the *Home* screen, allows the player to interact with the different markets in the game. The *Home* screen should display various information about the bank controlled by the player, open information about other banks, and the general state of the economy. As in the current stage, we focused more on the main architecture of the game and, as it was suggested on the M6 review, on the underlying simulation, this screen is currently empty. For this reason we do not discuss it here. In the next sections, we describe the other four screens that allow the player to interact with the markets.

4.2 Shares

The *Shares* screen (see Figure 4) is where the player can post orders on the stock market in the game. (S)he can decide to buy or sell shares, and manage the currently posted orders. At the top of the screen, there are widgets that are the same on any of the screens. On the right side,

there is a calendar that shows the current date in the game. In the middle there is a box containing general information about the bank controlled by the player, such as the bank's balance, equity, etc. Above this box are the buttons that are used to switch between the screens. Finally on the left, there is a special widget players can use to allocate certain percentage of their excess money to providing commercial loans, interbank loans, and buying shares. The corners of the triangle represent the three forms of investment, and the red dot shows the current selected division. If the red dot is dragged closer to one of the corners, a higher percentage of the excess money will be allocated to the corresponding investment form.
Below these widgets is the stock-market specific part of the screen. On the left side, the *'Player's Portfolio'* table lists the shares currently owned by the player. By selecting one of the owned shares and clicking on the *'Sell Order'* button, the player can specify the amount and limit price of the given share to be sold, and post a sell order on the stock market.

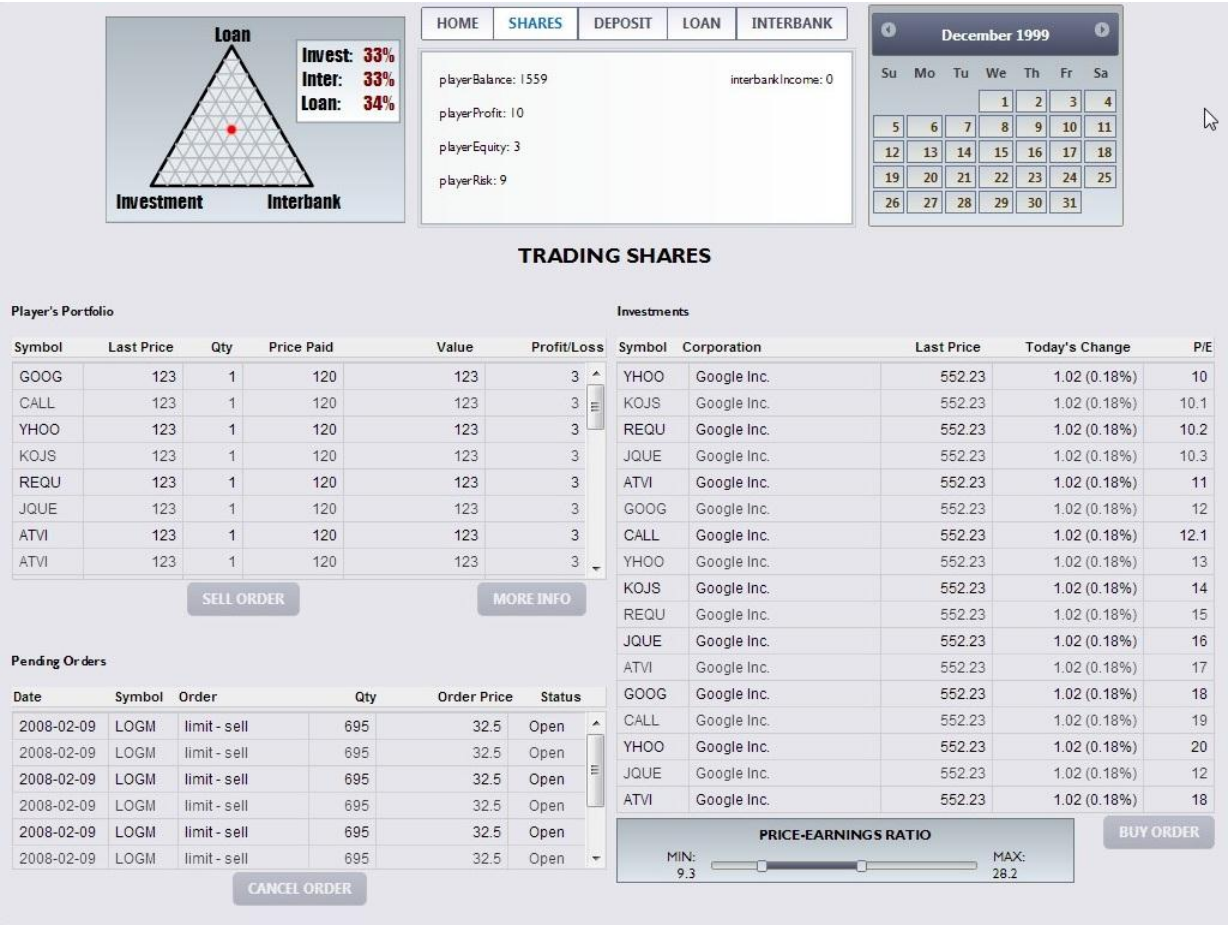

Figure 4 The Shares screen

Bellow the player's portfolio, the *'Pending Orders'* table lists all the orders currently submitted by the player to the stock market. By selecting an order and clicking the *'Cancel Order'* button, the order can be withdrawn from the market.
The right-hand side of the *Shares* screen is mainly occupied by a table called *'Investments'* that lists the available shares on the market. Here the player can browse the shares and by selecting one and clicking the *'Buy Order'* button, (s)he can provide an amount and a limit-price and post

a new by order on the market. What is displayed in the *'Investments'* table is constrained by the *'Price-Earning Ratio'* slider below the table. The table displays only the shares with a P/E value between the minimum and a maximum P/E value set on the slider.
In general, this screen allows the player to interact with the stock market in the game by posting new buy or sell orders, and by cancelling currently pending orders.

4.3 Deposit

The deposit screen (see Figure 5) is used by the player to acquire new deposits by posting new orders on the deposit market. The player can select a new deposit interest rate on the slider at the bottom of the screen, and by clicking the *'Submit'* button, a new deposit order is sent to the market. Deposit orders do not have a size parameter (it is automatically set to a very large number) as banks do not have a limit on the amount of deposit they can take. Deposit orders (similarly to other orders) stay active in the market until withdrawn; therefore players will always have an active deposit offer. When a new deposit order is sent by the player, the previous one is removed from the market after the new one is submitted.

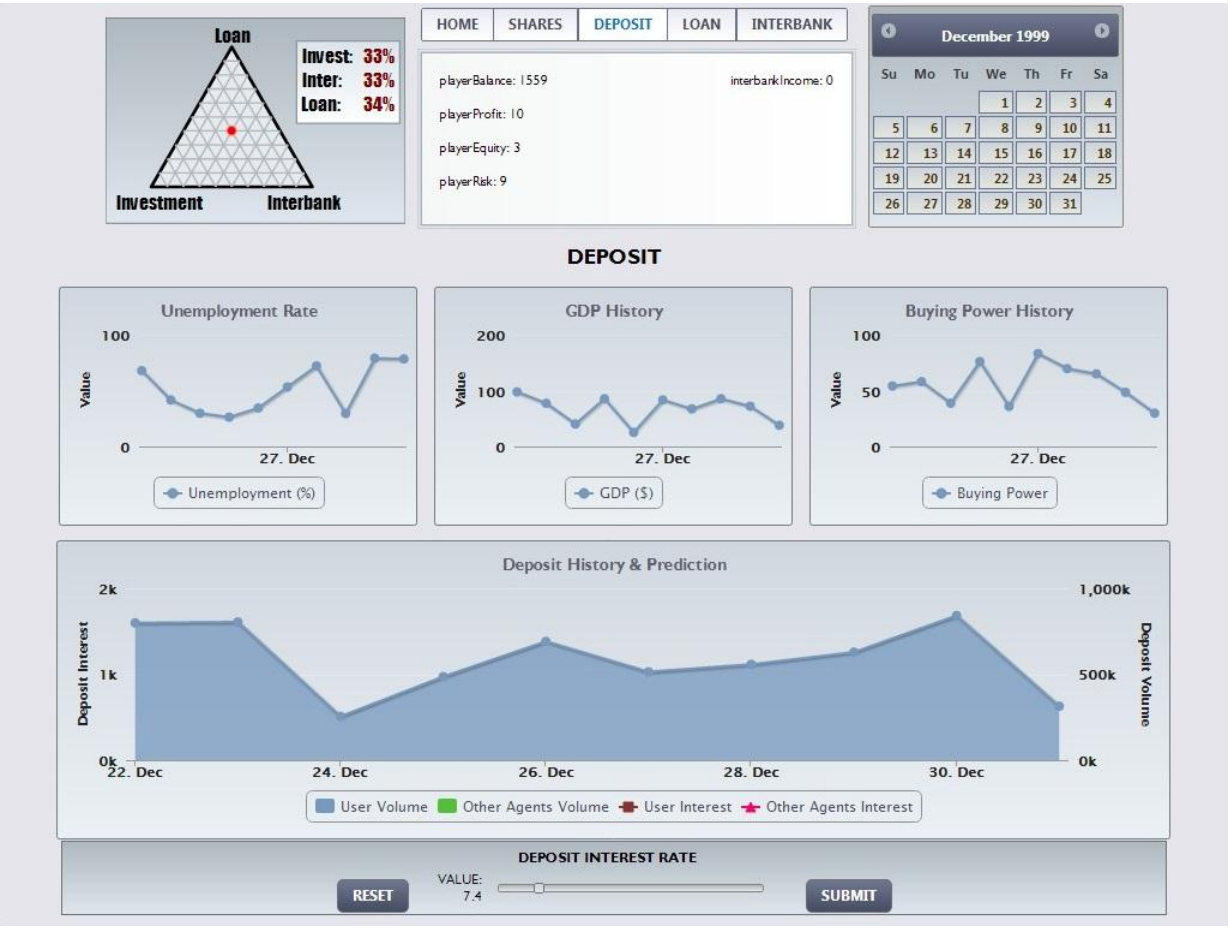

Figure 5 The Deposit screen

The displays in the middle of the screen show general statistics about the state of the economy, specifically the history of the unemployment rate, the history of the GDP, and the buying power

of households. Below these smaller charts, the bigger chart shows the history and short prediction of the deposits in the bank of the player, as well as aggregate values of the concurrent players. On the chart, the player can switch on and switch off the different data series (deposit volumes and interest rates). E.g. in Figure 5, only the history of the deposit volume of the player is displayed. All these charts are displayed to the player to help in deciding the proper value for the deposit interest rate of his/her next deposit order.

4.4 Commercial Loan

On the *Loan* screen (see Figure 6), the player can submit new loan offers for firms. With one submission two orders are sent to the commercial loan market, one for 'risky' and another one for 'safe' firms. (And at the same time, the previous two orders are removed from the market if they were not yet matched). In the bottom-right corner of the screen, the user can set the interest rates and the volumes corresponding to the two orders. Furthermore, with the *'Acceptable Risk'* slider, (s)he can set the threshold of riskiness. In the bottom-left corner the table provides information about the firms in the game, displaying among others the riskiness as computed by the simulation model. The red/green coloring of the table shows which firms are categorized as risky and as safe by the current setting of the threshold.

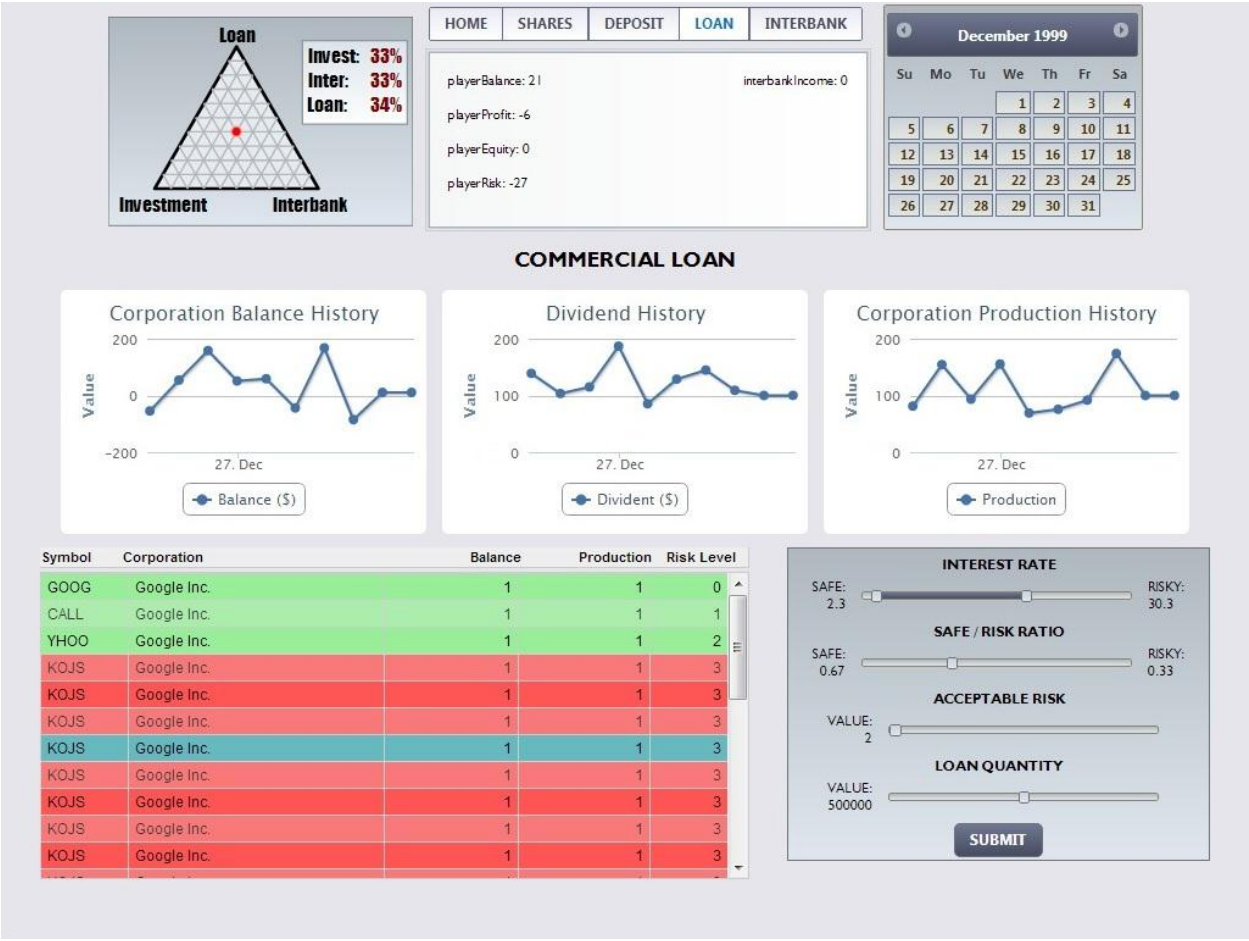

Figure 6 The Commercial Loan screen

Above the table, in the middle of the screen, small charts display public information about the firms selected in the table. Here players can get information about the individual firms, or groups of firms (if more than one firm is selected), or in general about the economy as a whole.

4.5 Interbank

The *Interbank* screen (see Figure 7) is used by the players to give and receive loans to and from other banks/players. Hence the screen has two sides, one for lending, and another one for borrowing. The user can change between the sides by clicking the small *'Lend'* and *'Borrow'* tabs at the side of the screen.

On the *Borrow* screen, players can submit buy orders to the interbank loan market. At the bottom of the page, they can select the amount and the interest-rate limit of the loan to submit, and they can press the *'Ask Loan'* button to send the request. The chart in the middle of the screen shows the history of the player's interbank loans with volume and interest rate, as well as the history of the total volumes (of interbank loans) and the average interest rate.

Figure 7 The Interbank Borrow screen

The *Lend* screen (see Figure 8) allows the player to submit sell orders to the interbank loan market. This screen is rather similar to the screen where the player offer loans to firms, except that here the list and the charts contain banks and not firms. That is, the table in the bottom-right

corner lists the partner banks categorized as ‘safe’ (green) and ‘risky’ (red) banks. When a bank is selected in the table, information about this bank, the history of its balance, profitability, and leverage, is displayed on the small charts above the banks. If more than one bank is selected, the charts display average information.

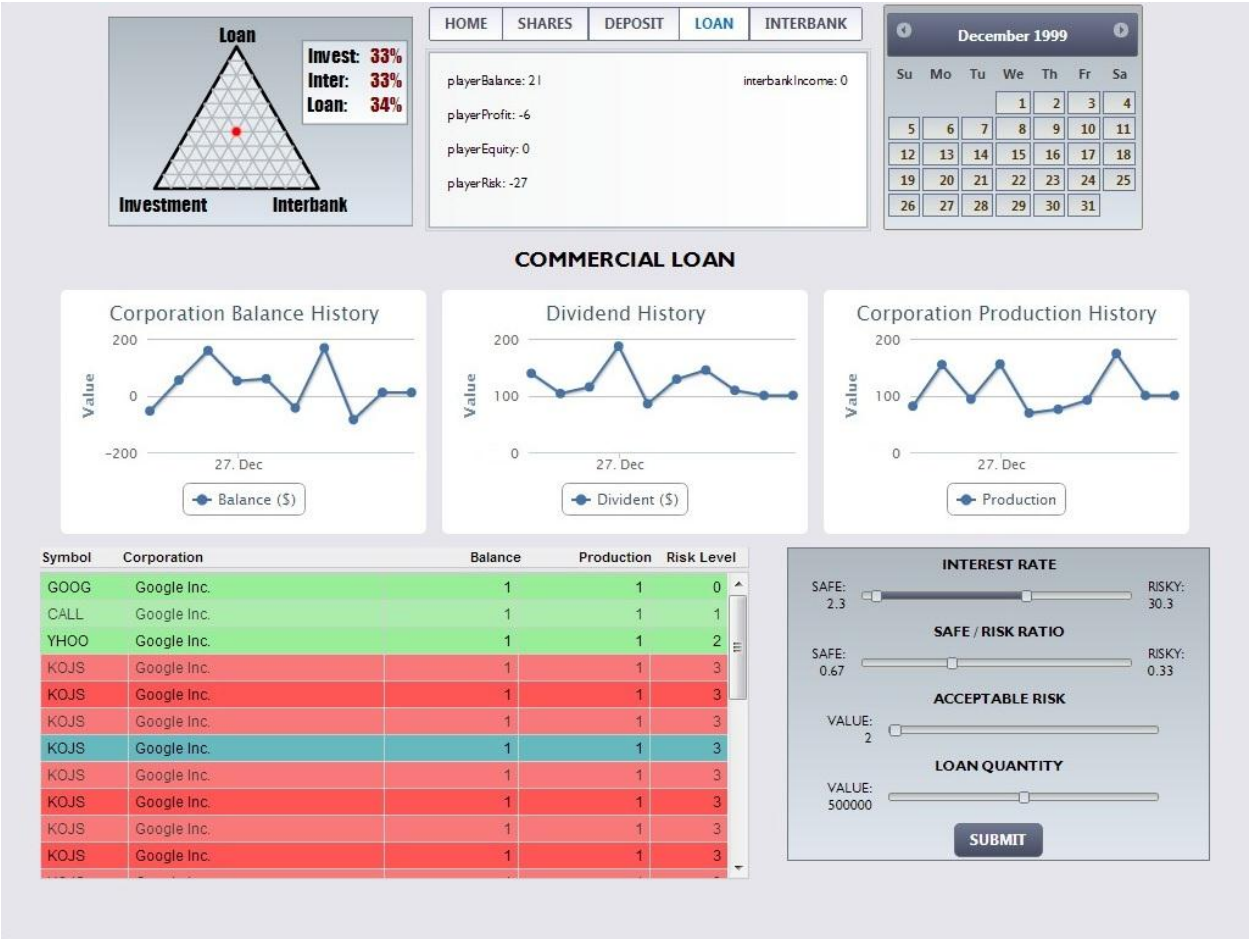

Figure 8 The Interbank Lend screen

Similar to the commercial loan screen, the player can set two volume and two interest rate values, one for the ‘risky’, and one for the ‘safe’ banks. When (s)he clicks the *‘Give Loan’* button, two orders are sent to the interbank loan market, replacing the two previously submitted orders. The *‘Acceptable Risk’* slider can be used to change the risky/safe categorization of banks on the left.

Using this two screens, the *Interbank Borrow* and *Lend* screens, the player can manipulate his/her orders on the interbank loan market, and thereby give and take loans to and from other banks/players.

All the screens described in the previous paragraph facilitate the interaction of the players and the underlying models. This interaction happens largely through the four market defined in the simulation model, the deposit, the commercial loan, the interbank loan, and the stock market. In addition to allowing the player to submit orders to these markets, the screens try to provide essential information to the player to make him/her be able to make decisions. Summaries of the economy, e.g. GDB, buying power of households, etc., important statistics of the firms and

banks are displayed to the user to facilitate decision making. Currently, information that seemed to be important is displayed on the screens, but this will surely change in the future as we gain more experience in what is really important to make the right decisions.

5 Summary

In this document, we described the first release of the CRISIS on-line game. In the first twelve month of the project, we designed and developed a multi-player browser-game that is based on an integrated macro and financial simulation model. The game allows the players to control banks with the goal of obtaining as much profit as possible. The game was mainly developed in WP5, with great help of WP2 and WP3 partners in the design of the game. The underlying simulation was and is mainly developed in WP8.

In this document, we described the current state of the game in a sequence of three sections. First we described the underlying simulation, and discussed how the Mark I model has been integrated with the financial markets of the base simulation library. Then we shortly introduced the game portal, and showed how games can be configured and started. Finally, we presented the game UI in its current state, and described what actions the players can perform, and how it affects the simulation.

In its current state, the game software is ready to be used within the project to experiment with the underlying simulation, to learn how it behaves, and to see which player strategies can lead to interesting games. In the following months, we will devote time to play with the game to gain experience with it, and to see how we can improve it. The graphical design of the game and the game portal could be improved, and the amount and type of information displayed to the players will most probably also change. Behind the scenes, the underlying simulation will improve too, as the model develops within WP8. What we can consider to be (close to) final at this stage is the design and implementation of the machinery that runs the simulations and facilitates the interaction between the players and the simulation. The part of the portal that initiates the game, the game engine and the game adaptor components had been carefully designed and implemented for this first release of the game. Now that this basic machinery is in place, we can concentrate on improving the gameplay and the game experience.