

**Settle And Destroy (SAD)**  
**Group 13**  
**Jonas Wikberg**  
**Christofer Hjalmarsson**  
**Daniel Westerberg**  
**Saul Amram**  
**André Sikborn Erixon**

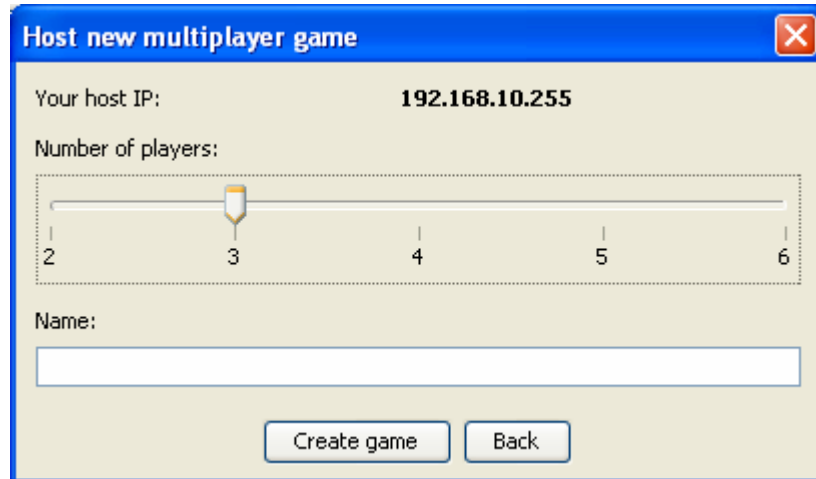
1.

2.

3.

## 4. Graphical user interface

### 4.1. Host new multiplayer game



#### 4.1.1. Names of the controls and fields

- labelYourHostIP
- labelIP
- labelNumberOfPlayers
- sliderNumberOfPlayers
- labelName
- textFieldName
- buttonAction
- buttonBack

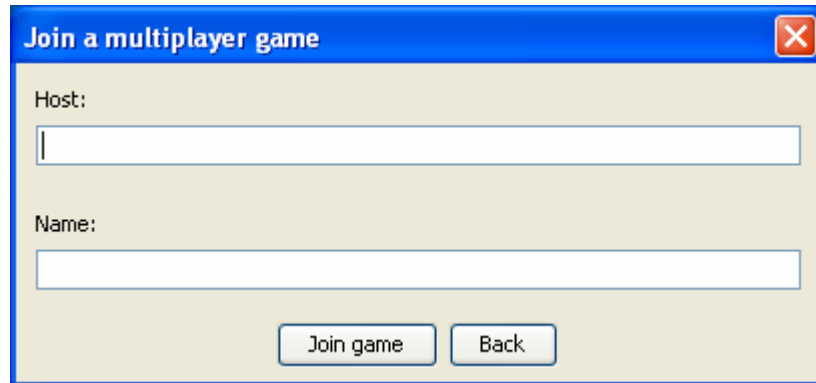
#### 4.1.2. Events, methods, or procedures that cause that form to be displayed

- `new HostGameDialog(JDialog).setVisible(true);`

#### 4.1.3. Events, methods, or procedures triggered by each control

None are predetermined. Events are added from external actors (through listeners) such as the window that opens this window.

## 4.2. Join a multiplayer game



### 4.2.1. Names of the controls and fields

- labelHost
- textFieldHost
- labelName
- textFieldName
- buttonAction
- buttonBack

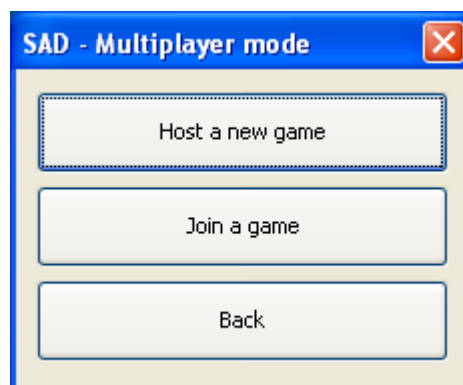
### 4.2.2. Events, methods, or procedures that cause that form to be displayed

- `new JoinGameDialog(JDialog).setVisible(true);`

### 4.2.3. Events, methods, or procedures triggered by each control

None are predetermined. Events are added from external actors (through listeners) such as the window that opens this window.

## 4.3. Multiplayer mode menu



### 4.3.1. Names of the controls and fields

- buttonHost
- buttonJoin
- buttonBack

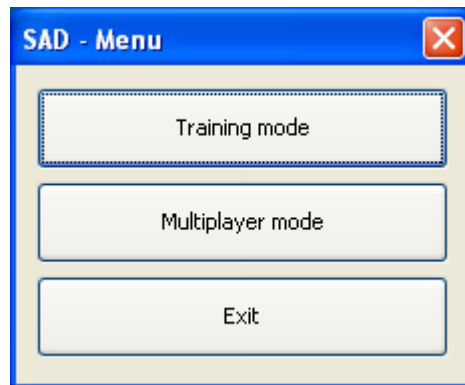
#### 4.3.2. Events, methods, or procedures that cause that form to be displayed

- `new MultiplayerModeDialog(JDialog).setVisible(true);`

#### 4.3.3. Events, methods, or procedures triggered by each control

- `buttonHost - buttonHostActionPerformed(ActionEvent evt)`
- `buttonJoin - buttonJoinActionPerformed(ActionEvent evt)`
- `buttonBack - buttonBackActionPerformed(ActionEvent evt)`

#### 4.4. Start menu



#### 4.4.1. Names of the controls and fields

- `buttonTrainingMode`
- `buttonMultiplayerMode`
- `buttonExit`

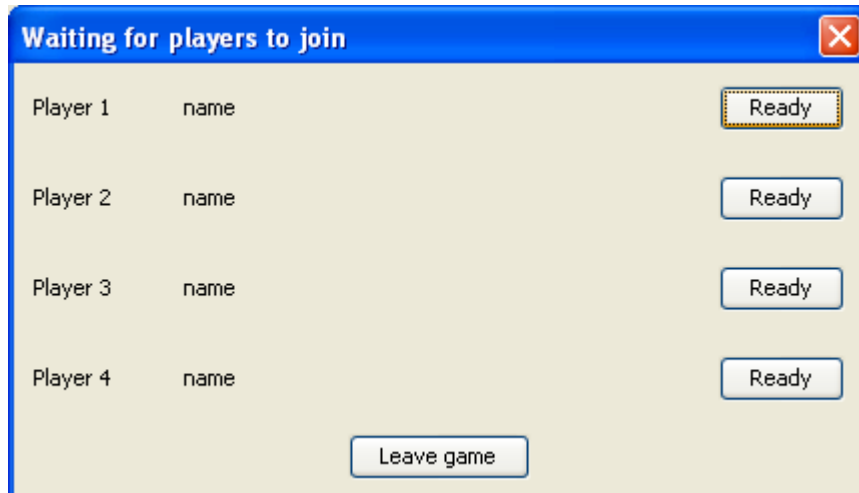
#### 4.4.2. Events, methods, or procedures that cause that form to be displayed

- `new StartMenuDialog().setVisible(true);`

#### 4.4.3. Events, methods, or procedures triggered by each control

- `buttonTrainingMode - buttonExitActionPerformed(ActionEvent evt)`
- `buttonMultiplayerMode - buttonMultiplayerModeActionPerformed(ActionEvent evt)`
- `buttonExit - buttonExitActionPerformed(ActionEvent evt)`

## 4.5. Waiting for players to join



### 4.5.1. Names of the controls and fields

- labelPlayer
- labelName
- buttonToggleReady
- buttonOne

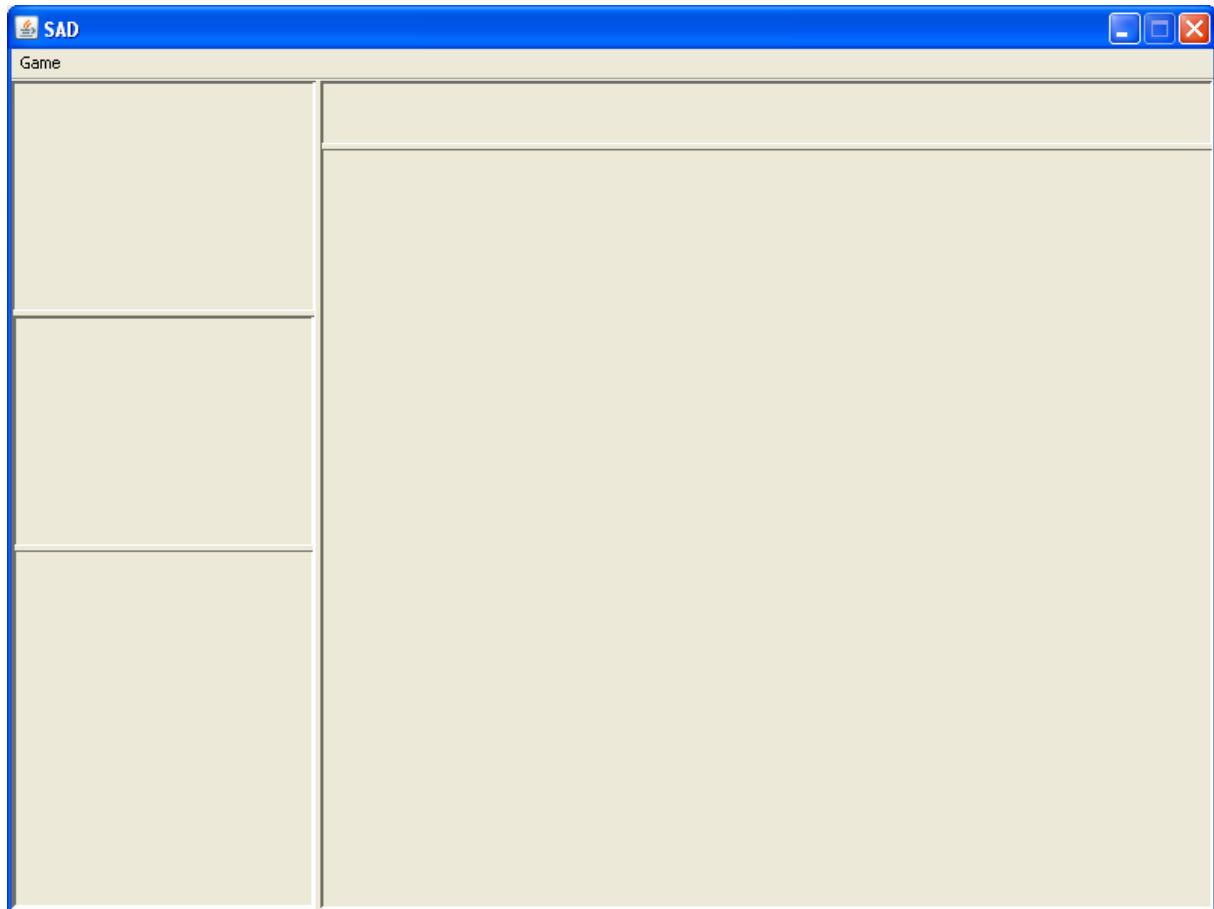
### 4.5.2. Events, methods, or procedures that cause that form to be displayed

- `new MultiplayerWaitingReadyDialog(JDialog, int).setVisible(true);`

### 4.5.3. Events, methods, or procedures triggered by each control

None are predetermined. Events are added from external actors (through listeners) such as the window that opens this window.

## 4.6. Main game window



### 4.6.1. Names of the controls and fields

- menu
- menuGame
- jSplitPane1
- jSplitPane2
- jSplitPane3
- jSplitPane4
- panelBuilding
- panelSelection
- panelInfo
- panelMap

### 4.6.2. Events, methods, or procedures that cause that form to be displayed

- `new MainWindow().setVisible(true);`

### 4.6.3. Events, methods, or procedures triggered by each control

None are predetermined. Events are added from external actors (through listeners) such as the window that opens this window.