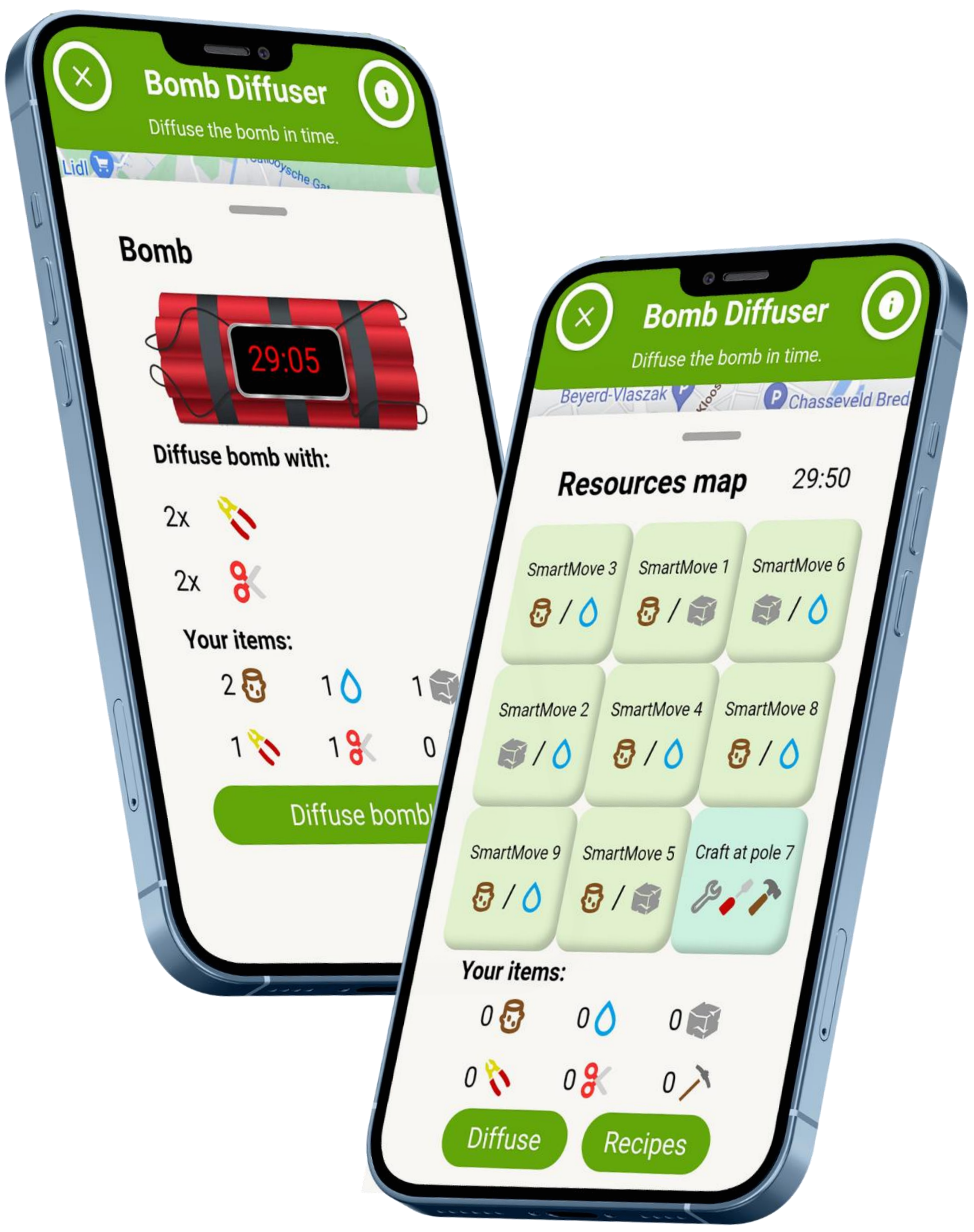


SmartMove bomb diffuse game

Advisory report

Group 2

Iris Roemermann, Niels Lazaroms, Hugo Boulouard and Gabin Fily

Table of contents

Product description

Our team has created an exciting bomb diffusion game that aims to offer you a captivating gaming adventure seamlessly combined with physical activity. Bomb Diffuser sets you on a thrilling mission: defuse a bomb within a time limit. The game provides diverse experiences with both solo and multiplayer modes.

The objective is to defuse a bomb before time runs out, whether you're going solo or challenging friends in multiplayer mode. Interact with SmartMove Poles along walking routes, strategically collecting resources. Then, you'll need to use your resources to craft the items necessary to defuse the bomb. In single-player mode, your goal is to defuse the bomb before the timer expires, while in multiplayer, it's a friendly race against both time and your friends to achieve successful bomb diffusion. Gather resources, craft items, and emerge victorious before the countdown reaches zero.

Our game is an outdoor adventure that encourages users to move and interact with their surroundings. By combining the thrill of bomb deactivation with resource collection, we've created a unique gaming experience that promotes physical activity and entertainment. Whether you're playing solo or competing with friends, our app inspires users to stay active and enjoy a playful fitness experience.

Product features

What can be done with the product?

When starting the game, the user can either play singleplayer or multiplayer. The singleplayer is just playing alone against the time and with multiplayer, the user can play against friends. With multiplayer, the user can set their nickname for the game and select a lobby room to join. If everyone is ready, the game will begin.

Everyone starts with 0 resources and 0 items. You see a resource map, showing at what SmartMove you can get what resources, existing of water, wood and stone. These resources are randomly generated and you can’t go to the same SmartMove after 1 minute again. With the collected resources, the user can craft certain items such as scissors, pliers and a pickaxe at SmartMove number 7. These items are needed to diffuse the bomb, which is located at SmartMove number 10. The user has 30 minutes to diffuse the bomb, else you’ll lose the game and need to restart.

The items that are needed to diffuse the bomb are also always randomized, so you won’t play the same game twice and it has endless playing time.

The user will see different pop ups, either what resource they collected, what item they crafted or if they don’t have the right resources/items.

If the user diffuses the bomb in time, they win the game. The winning screen shows the time the user took to diffuse the bomb.

With multiplayer, you play against everyone, so who collects and crafts the right items before the bomb explodes in time will win. In multiplayer, only one person wins, since you don’t play as a team.

What works well?

The interaction between our product and the SmartMoves work really well, since we used the provided code from Embedded Fitness. Now when an user interacts with a SmartMove, the app will respond to it.

The items that are needed for the bomb get randomized every new game round, just like the resources map. The resources the user collects when interacting with a SmartMove also gets randomized every time, so it’s really a 50/50 chance.

The product has a timer of 30 minutes to diffuse the bomb, when the user runs out of time, it’s game over. If the user crafts the right items and diffuses the bomb in these 30 minutes, they win. At the winning end screen, the user will see the time they took to diffuse the bomb.

With every action, the user will get a clear response by showing them a pop up. The pop ups will be shown when collecting a resource at a SmartMove, crafting an item and also when the user don’t have the right resources to craft or not the right items to diffuse the bomb.

The checking if the user has the right resources to craft and the right items to diffuse the bomb works also very well. This is done with some if-statements that checks the values of what the user has and will either be true or false.

The ‘my items’ count section also works very well, if the user collects a new resource or crafts a new item, the ‘my items’ section immediately gets updated.

What needs improvement?

There are a few minor things that still needs improvement.

First thing that can be improved is that now the resources that can be collected at a pole is hardcoded, so it’s always the same 2 resources and never changes. This could be improvement by making the functionality that the resources that can be collected are always random. This also counts for crafting items with resources, the user always needs the same amount of resources to craft a specific item.

Second thing that can be improved is to randomize which SmartMove is the bomb and which SmartMove is the crafting table. Right now, the bomb is always 10 and the crafting table is always located at number 7. This is set in the JSON file we’ve made. This can be improved by making the logic to randomize these locations.

A third thing that can be improved is to add different difficulties in the game. Right now, the user always has 30 minutes to diffuse the bomb and the amount of resources needed to craft stays the same. If you want to make the game easier, the time could get longer and the needed resources count can be lower. The hardest difficulty could be that you need a lot of resources to craft while having less time to diffuse. These are all optional things that could make the gameplay different and more customizable.

Another thing that could be improved is to make the game more adjustable to other places with SmartMoves. The game now has the SmartMoves from Breda and their ID’s hardcoded, but if users want to play the game at another location that has SmartMoves, the game should be able to adjust to these SmartMoves. This way more people over the country will be able to play our bomb diffuse game.

Product usage

Compatibility

Safari iOS 11.3 or higher

Android 4.4 or higher

How to install our product

You can clone our Git repository by running this command at the location you want to have the code:

```
$ git clone https://github.com/xRedRosee/SmartMove-bomb-diffuse-game.git
```

You have to make sure you're now in the bomb diffuse folder, if not then go into this folder with

```
$ cd smartmove-group2-bombdiffuser
```

Then, you need run the following npm command at the root of the project **and** in the backend folder.

```
$ npm install
```

By running this command, you will install all of the dependencies that are listed in the projects' package.json.

To start the application for development, you'll need to run this npm command at the location:

```
$ npm run dev
```

This will start up the Vite server.

If you want to run the server locally, you will need to open another terminal. Enter first this to make sure you're in the right map:

```
$ cd .\backend\
```

And then to start the server:

```
$ node .\server.js.
```

To put your application in production, run the following npm command at the root of the project:

```
$ npm run build
```

This creates a build directory with a production build of the application, located at the dist folder.

Then, you will find all the generated production files and assets in the dist folder. You will need that to deploy and run your application.

We created a new Git with only the server to host it. We used Render (https://render.com/) but you can use any hosting service that supports Node.js server. Here is the Git: https://gitlab.com/Gabin29280/smartmove-group2-prototype2/tree/main

Future plans

Advise code

Adding resources

We advise to add resources or change a few to make it more logically. If you want to add more resources or tools, than you should also think of a new solution off displaying "your items". You can add new items, and they will be showed in the list as well properly. But the thing that will happen is that the "Go back" button will disappear.

Randomize bomb and crafting location

We advise to make sure that the bomb and the crafting location can also be randomized between all the available SmartMoves. Now they are hardcoded at SmartMove 7 & 10, but we think if you randomize these it will be more fun for the user.

Hardcoding SmartMove ID's

In our game the SmartMove id's are hardcoded, but we advise to think of a different way where you don't do this. If you get rid of this hardcoded part than it will be easier to use it add different locations.

Making the craft recipes random

Now there are a few standard craft recipes, but to make the game more repayable we advise to make more random craft recipes, or the amount of items can be randomized. With this it will vary from game to game how many resources you need for a specific item.

Create room functionality

If you want that the users will be able to create their own room, we advise to take consider implementing the create room functionality. In this functionality the user will be able to create their own room. The user that creates the room will be the host. The host will be able to start the game and kick other users. A possible design for this can be found below.

Create trading functionality

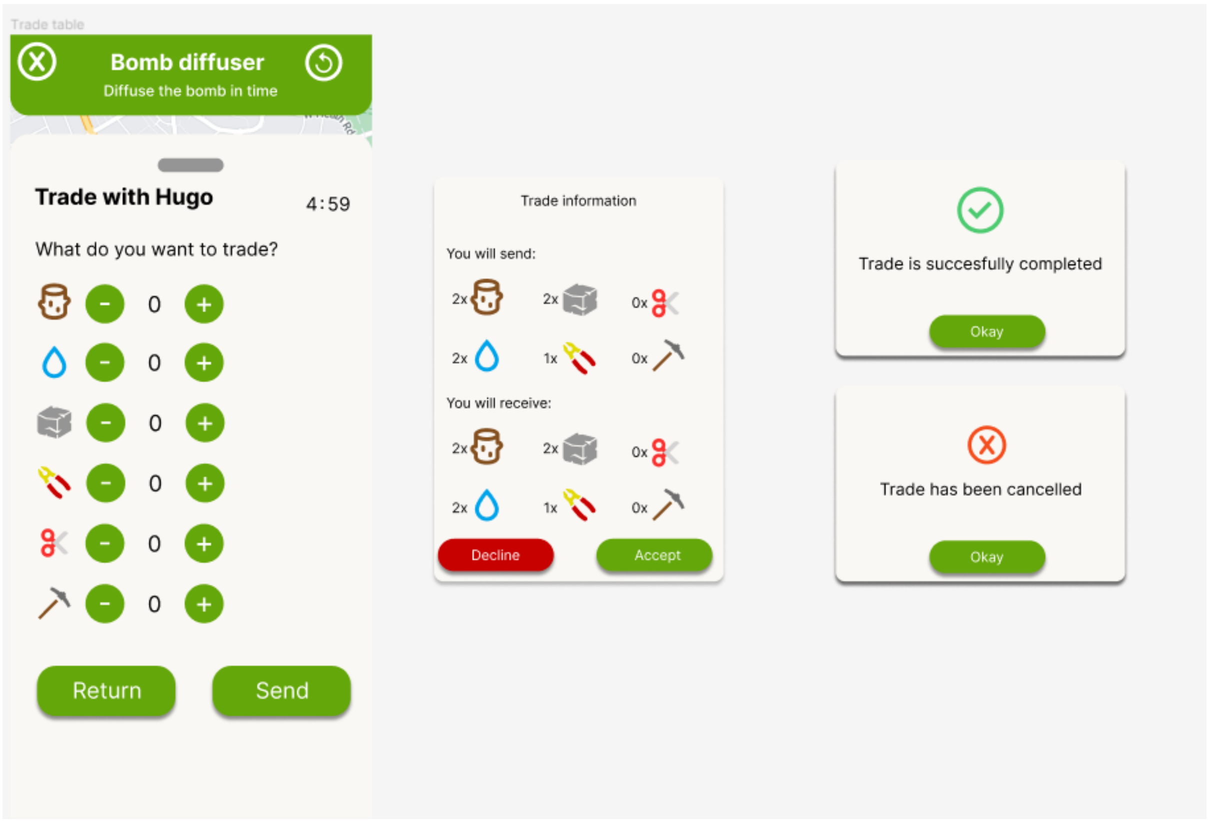
If you want to implement the multiplayer game where the users should work together to win the game, you could consider implementing a trading functionality. In this trading functionality the users should work together to get the correct items. Because in this mode 1 user can collect a specific item and the other user the other needed resources. So, they should work together to craft items. A possible design for this can be found below. To trade the user should also interact with a SmartMove. The concept of this is the same as the bomb, and the crafting but its just a different SmartMove.

Advise design

For the design we could advice a few minor changes. These changes depend on the features that will be implemented.

Trading design

If you want to implement the trading part for the multiplayer game, then we have already thought of a possible design which can be found in our Figma. In this design we made sure that the user will be able to trade, select items to trade & the user can choose to accept or decline the trade request.

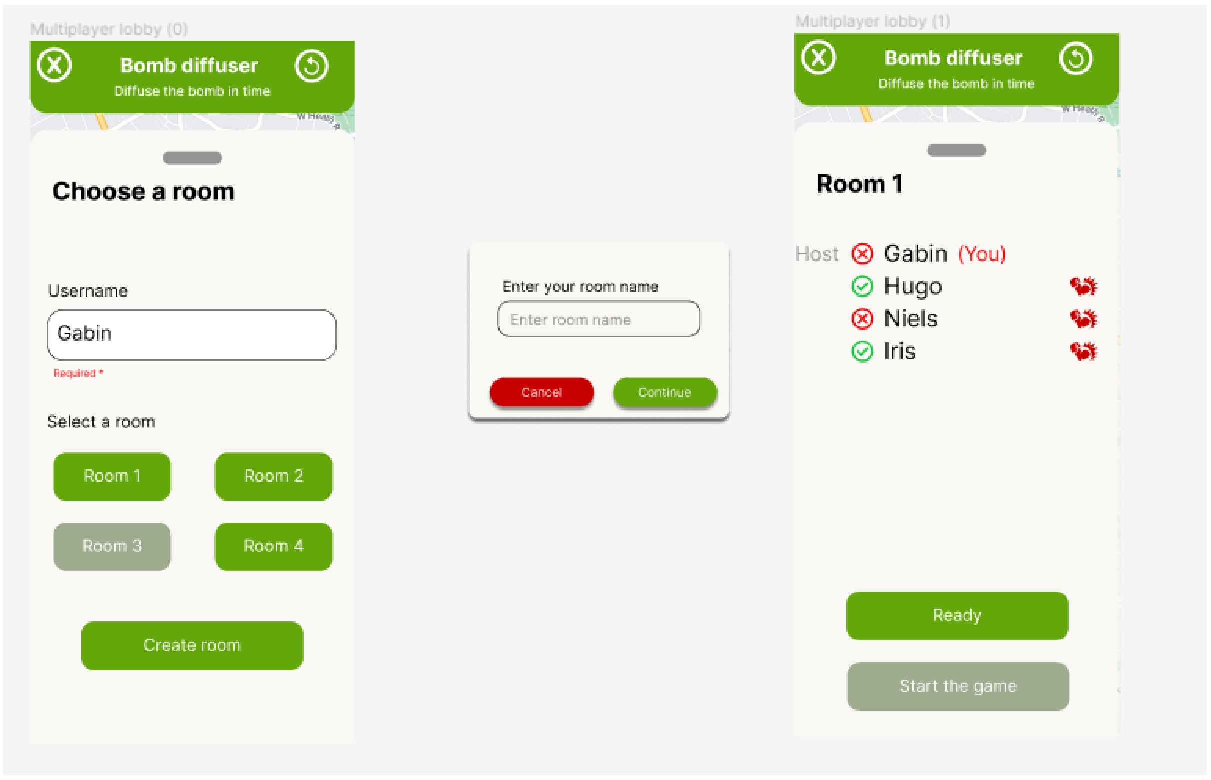


This is a possible design for the trading design.

Create room / lobby updates.

If you want to implement a host function for the rooms, we would also advice to work on a create-room button. With this button you will make sure that the user will can create their own room, when people can create their own room, you also make sure that there can be hosted more multiplayer games at a time.

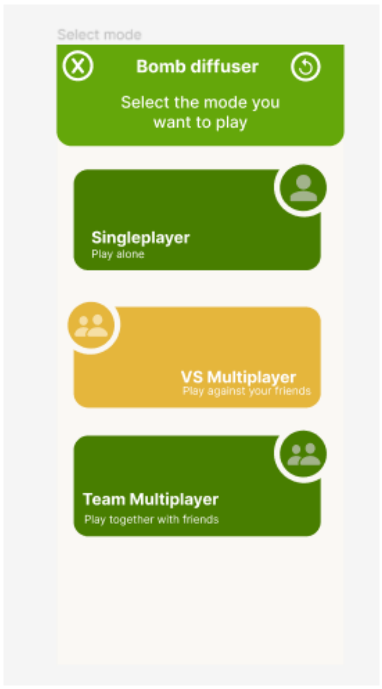
When there is a host, you can also implement a kick button. With this button you will be able to make sure that the unwanted users can be kicked out of your own room. In the image you will find the design ideas that we had in mind.



Possible design solution create room / lobby updates

Select mode screen

In case that you choose to implement the trading part, we advise to consider changing the select mode screen, where you add a new button which says, “Team multiplayer” and change the text on the multiplayer button to “VS multiplayer”. With this you will be able to choose either one of the two modes. In the image you see a possible design solution that we thought of.


Select mode
Bomb diffuser
Select the mode you want to play
Singleplayer
Play alone
VS Multiplayer
Play against your friends
Team Multiplayer
Play together with friends

Select mode screen possible design.