

# Minecraft project essay sample



**ASSIGN  
BUSTER**

In order to reinforce the knowledge acquired in class we were asked to develop a software system for a game. The aim of this project is to introduce the design of a cloud based game for children; the trigger for the problem is based on Minecraft. Methods of software design were used to meet the criteria and implement the system. The first step for designing a software system is to identify its actor; in Minecraft game two actors were identified, these actors somehow interact with the system, in the following lines the actors will be presented and their interaction with the system will be shown in form of use case. Actors:

Players

Administrator

Flow of events:

The use case starts when player goes to the login UI.

Basic Flow:

1. Player inputs username and password and system identifies user.
2. System authenticates user and a second screen with manageServer is displayed.
3. Player chooses in which server he wants to play and the system take the player to the preGame screen.
4. Player chooses if wants to start a newGame or loadSavedGame.
5. System takes player to the game
6. Player craftItems
7. Player saveGame
8. Player quitGame

### Alternate Flow

1. Player inputs the wrong username or password. It goes back to Step 1.
  2. Player does not have an account. Player goes to createAccount and after goes back to Step 1.
  3. Servers are not working. Player is back to Step 2.
  4. Player does not have a saveData. System only displays newGame option.
  5. Player does not have the latest version of the game. System request player to update game and after the game is updated player is directed to Step 2.
- Special Requirement:

Internet Connection to access the save data on the Cloud and to play using online servers. Pre-conditions:

Player needs to have/create an account before start with a game. Post-conditions:

If use case successful, player will be able to choose between servers, game mode new game or load saved data and also be able to play the game, craft items, save and quit it and also update the game when needed.

### USE CASE DIAGRAM

In the diagram below is it possible to check how the actors interaction with the Minecraft System

### FIGURE 2

### CONCEPTUAL CLASS DIAGRAM

The class diagrams show the attributes, operation, class name sand also the associations that in this example is a bi-directional association with the multiplicity in each point where it can be: 0.. 1Zero or one

1. 1One only

0.. nZero or more

nZero or more

1.. nOne or more

FIGURE 3

GLOSSARY

FIGURE 4

SYSTEM SEQUENCE CLASS DIAGRAM

FIGURE 5

CONTRACT

FIGURE 6

COLLABORATION DIAGRAM

The UML Collaboration diagram bellow was created in order to model how the objects involved in the Minecraft scenario interact with each other and their job is to instantiate the class of the system. The objects are connected by links; the links represent an instance of the association between the classes. The messages are exchanged between the objects to specify what the steps of the interaction are. Communication explained step by step:

Step 1 – players enter their credentials in order to login

Step 2 – the system verify if the credentials are correct

Step3 – if credentials are correct the game starts

Step4 – system message credentials are not correct

Step 5 – start of the game is denied

Step 6 – device writes on the screen create an account

<https://assignbuster.com/minecraft-project-essay-sample/>

### COMPONENT DIAGRAM

The structure of the software is modelled and presented in the component diagram bellow; the diagram shows the dependency existent between the components.

### DEPLOYMENT DIAGRAM

The deployment diagram shows the physical relationship among software and hardware components; as required for the Minecraft project we can found the nodes and the communication association existent between them, the bellow diagram gives us an idea on how the nodes and how they communicate and what links and dependency they have, we can see the links between nodes and the dependency of the components.