ANKARA UNIVERSITY

ENGINEERING FACULTY

DEPARTMENT OF COMPUTER ENGINEERING

INTERNSHIP REPORT

Gameplay Programming Using C#

Yeter Tuğba Çetin

16290085

24.06.2019-22.07.2019

ABSTRACT

This report has been prepared to include and explain the internship period I have worked during the summer term.

What I have done during my internship was mostly to complete the tasks and fix the bugs assigned to me by my supervisor and other programmers in my team. I used C# language and XML (Extensible Markup Language) while working on assignments that I had. My purpose was to grasp the fundamental principles of game development, to develop myself in C# and to learn XML that I had not used before. I have also used these languages on issues and errors that I have encountered to solve them efficiently.

In the end, I learned how to fix some possible bugs while developing the game. While correcting these errors, I learned how to create the solutions I found run within the project more efficient and faster.

INSTITUTION INFORMATION

Institution;

Name	: Taleworlds Entertainment
Department	: Campaign
Address	: Galyum Blok, ODTÜ Teknokent Ankara, Turkey 06531
Telephone	: +90 312 210 00 27
E-mail	: info@taleworlds.com
Web Page	: https://www.taleworlds.com

Taleworlds Entertainment is an official brand of İkisoft Software Company have been developing computer games founded in 2005 in Ankara.

The company’s first game is Mount & Blade which is an RPG (Role-Playing) game that is launched in 2008. Today, the company has launched its games on platforms other than PC such as XBOX and PlayStation.

Games launched by the company:

- Mount & Blade
- Mount & Blade: Warband
- Mount & Blade: With Fire & Sword
- Mount & Blade: Warband - Napoleonic Wars developed by Flying Squirrel Entertainment
- Mount & Blade: Warband - Viking Conquest

The company is currently working on the game project Mount & Blade II: Bannerlord.

Supervisor:

Signature:

TABLE OF CONTENTS

ABSTRACT..i

INSTITUTION INFORMATION..ii

TABLE OF CONTENTS...iii

1. INTRODUCTION..1

2. ASSIGNMENTS GIVEN BY THE SUPERVISOR..2

2.1. Agent Displacement Frequency in the Settlements.........................2

2.2. Leave Army Text...2

2.3. Leaving Behaviour Using Dungeon Door..2

2.4. Naming Inconsistency in Encountering Conversations……………..2

2.5. Id Refactoring...3

2.6. Food Notification Float..3

2.7. Incompabilityof Influence Between Notification and Overlay..........3

2.8. Printing Possible Battle Scenes...3

2.9. Party Speed Calculation Bug..4

2.10. Review Names in the Name Generator Records...............................4

2.11. Clamp the Relation For Appropriate Values of Relationships..........4

2.12. Conversation Fix...5

2.13. Appropriate Menu Display...5

2.14. Used Kingdom Banners Shouldn't Be Visible in Banner Editor.......5

2.15. Cheat Refactoring..6

2.16. Add a New Notification to the Default Notifications.........................8

2.17. Recruiter Cultures In Conquered Cities...9

2.18. About the Assigned but Rejected/Closed Issues.............................9

3. PROJECT ORGANIZATION SOFTWARES..10

3.1. Plastic SCM...10

3.2. JIRA...10

4.CONCLUSION...11

5. BIBLIOGRAPHY..12

1. INTRODUCTION

I completed my summer internship at Ikisoft Taleworlds Entertainment that is a game development company in Technopark, METU. It lasted four weeks and started on 24/06/19 and ended on 22/07/2019. During my internship, I was commissioned to handle the tasks and bugs that my supervisors assigned me in the game Mount & Blade II: Bannerlord.

In my report, I started by giving brief information about the company I worked in and what kinds of products they have. I then explained the issues I worked on adhering to the company's privacy policy. The sub-headings in the second part of the report are abbreviations of the problem headings sent to me through the software called Jira. As part of the privacy policy, I could not add any source code or software files to the report, so some of the code pieces I have written are included in the report as an example.

At the beginning of my internship, I learned about the use of software used in-house. I used Plastic SCM for the project organization, Visual Studio for code writing and JIRA for issue assignments. Then I solved the various issues that my supervisor gave me through Jira. During the process, I have examined the relevant parts of the project in detail to find out how to solve the assigned tasks efficiently. The explanations of these tasks are given in detail in the second part of the report. The details of the software I used for the first time are also included.

Following that, I made a sum in the “Conclusion” part of my work and what I have learnt during the internship.

I am very gratified to have the opportunity to work for Taleworlds Entertainment. Working in the company brought me a very satisfactory learning process. In addition to learning, I got a cursory idea of the challenges of getting involved in a great project and overcoming these challenges.

2. ASSIGNMENTS GIVEN BY THE SUPERVISOR

During my internship, I took part in the Campaign team of Mount and Blade II: Bannerlord, which the company is currently working on. The assignments that I have received from my supervisor during my time in this team, which is more concerned with the gameplay of the game, are explained in sub-headings in this section.

2.1. Agent Displacement Frequency in the Settlements

The positions and duties that will be performed by the non-playable characters of various statuses within the settlements have been updated and made more vivid by testing the values in the XML records. To do this, I updated the time intervals in the XML files more realistically according to the characteristics of the characters.

2.2. Leave Army Text

Some of the texts in the menus to leave the army during the attack and defence are updated. However, while correcting this problem, it was decided to rearrange the design that caused the menus to open. Therefore, a new issue is opened on the JIRA and assigned to the relevant person.

2.3. Leaving Behaviour Using Dungeon Door

When the user tried to exit the dungeon using the door, he was led to a menu on the main map instead of the current settlement’s streets. At the same time, one of the choices on the menu was not running correctly due to this problem. After finding the sections related to this behaviour in the code, I updated the conditions.

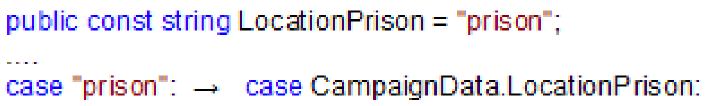
2.4. Naming Inconsistency in Encountering Conversations

The notifications contained information errors related to the player's presence. A similar situation of this misinformation was in a conversation the player had with the characters in the game. Related conditions have been edited and texts are updated with the given format.

2.5. Id Refactoring

Because Mount and Blade II: Bannerlord is a large project, accessing data from some files through a single class is of great importance for people working on the project for readability. For this reason, variables placed in a scattered structure are gathered beneath a particular class.

While I was editing, I updated the class again by including all other variables in a similar state that were not noticed in the project.



```
public const string LocationPrison = "prison";
....
case "prison":  →   case CampaignData.LocationPrison:
```

Figure 2.5.1: An Example of Variables

2.6. Food Notification Float

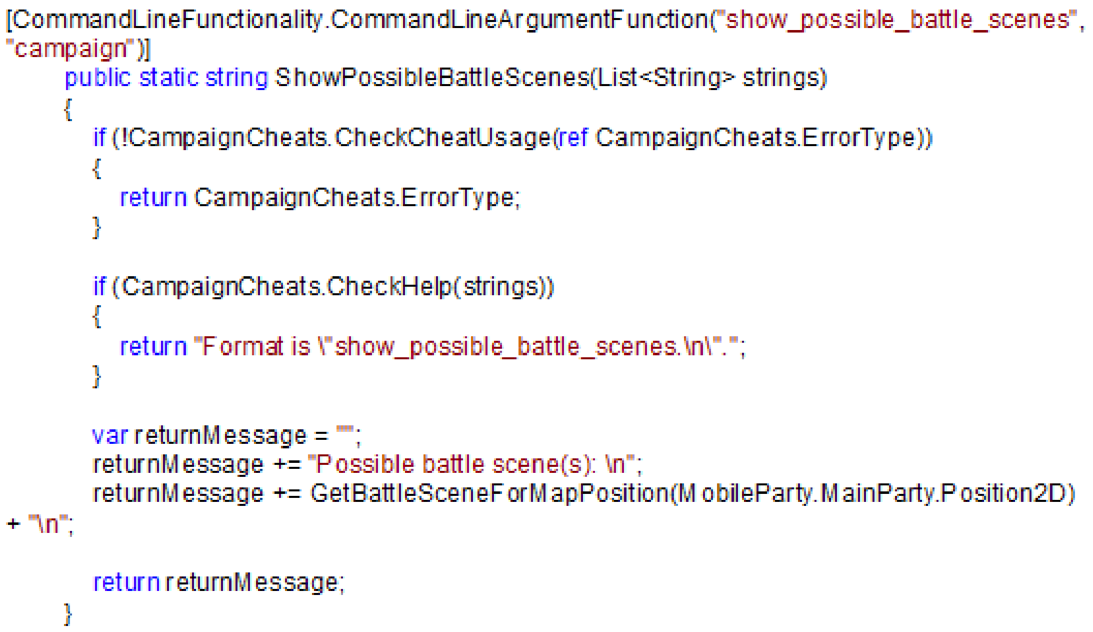
Some automatic calculations on the debug screen created for the user to follow the game caused image pollution. Therefore, the results of the related class were updated again in a more appropriate format. For this, I used the string formatting method in the C# library.

2.7. Incompabilityof Influence Between Notification and Overlay

I fixed the issue where the influence value contained in the main character's properties conflicts with values in other information fields on the screen. However, I noticed during the correction influence decreased to a negative value. It was undesirable that's why I corrected that issue too within the knowledge of my superiors.

2.8. Printing Possible Battle Scenes

A cheat has been written to list possible battle scenes that can be opened according to the area where the player is on the campaign map.



```
[CommandLineFunctionality.CommandLineArgumentFunction("show_possible_battle_scenes",
"campaign")]
        public static string ShowPossibleBattleScenes(List<String> strings)
        {
            if (!CampaignCheats.CheckCheatUsage(ref CampaignCheats.ErrorType))
            {
                return CampaignCheats.ErrorType;
            }

            if (CampaignCheats.CheckHelp(strings))
            {
                return "Format is \"show_possible_battle_scenes.\n\".";
            }

            var returnMessage = "";
            returnMessage += "Possible battle scene(s): \n";
            returnMessage += GetBattleSceneForMapPosition(MobileParty.MainParty.Position2D)
+ "\n";

            return returnMessage;
        }
```

Figure 2.8.1: Created Cheat for Printing Possible Battle Scenes

2.9. Party Speed Calculation Bug

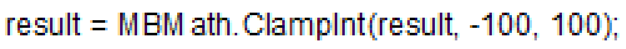
There was no change in speed when the main character in the game descended from his horse. I have moved the function that makes the necessary update to the correct area in the project. I then made the necessary adjustments following the review I received on improving the function and sent it to the relevant unit for testing through Jira.

2.10. Review Names in the Name Generator Records

During the beta test, some of the character names in the game are found to be incompatible with the theme and the whole game so the relevant XML files that records exist were examined and lists were arranged.

2.11. Clamp the Relation For Appropriate Values of Relationships

The minimum and maximum values of the relationships between the characters were rearranged. During the correction, some values were found to conflict. The aforementioned issue was noted for later evaluation to fully determine its design.



```
result = MBMath.ClampInt(result, -100, 100);
```

Figure 2.11.1: An Example for Clamping the Value From the Code

2.12. Conversation Fix

The given editing request was to enable an animation test speech prepared for use in developer mode to run the corresponding animations. However, since I learned that these animations were abandoned, I removed the relevant test section from the project.

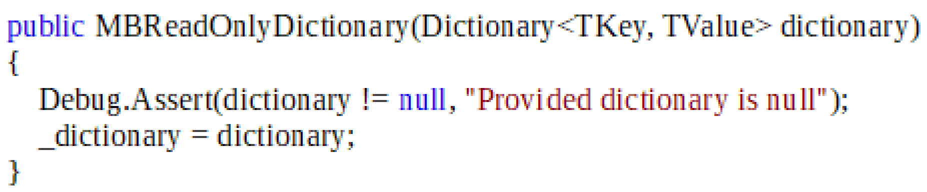
2.13. Appropriate Menu Display

It was seen that the relevant menus actually existed and were called correctly but that the opening of the menu was prevented before the desired situation in a different part of the project. This bug has been fixed.

2.14. Used Kingdom Banners Shouldn’t Be Visible in Banner Editor

In the banner customization that the user has access to within the game, the banner icons obtained from the XML file are edited via the assigned keyword to hide previously used ones.

When I submitted my change for review later, my supervisor gave feedback asking for the security improvement of the dictionary class that I used to solve the problem. I updated the existing dictionary to make the icons readable only by other classes and rearranged the files I used following the hierarchy reference in the document shared with me at the beginning of my internship for better readability.



```
public MBReadOnlyDictionary(Dictionary<TKey, TValue> dictionary)
{
   Debug.Assert(dictionary != null, "Provided dictionary is null");
   _dictionary = dictionary;
}
```

Figure 2.14.1: An Example Function from the Dictionary Class

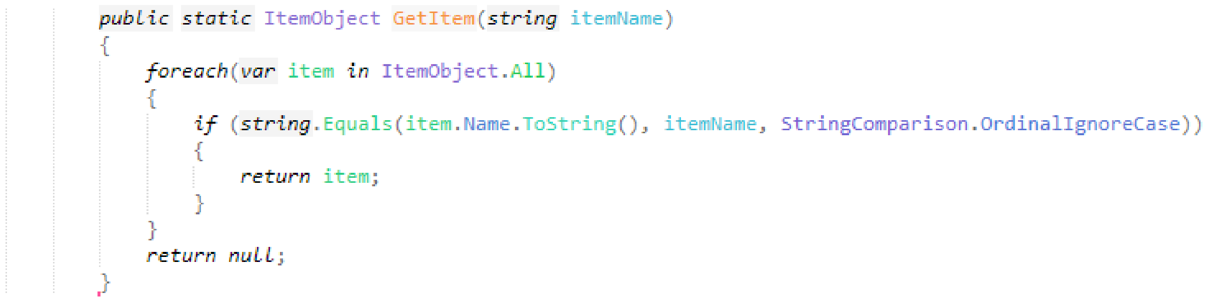
2.15. Cheat Refactoring

Defaults to cheat codes within the game, which are written to make it easier for players to play, have been refactored and some cheat codes are rewritten. New functions have also been added to make it easier to write cheats that will be added later. (Figure 2.15.1)

Below is an example of the code I wrote for the task in Figure 2.15.2 and Figure 2.15.3. Also, the names of the cheats I intervened are given in Table 2.15.1.

Table 2.15.1: The Key Codes for Using the Cheats in the Game

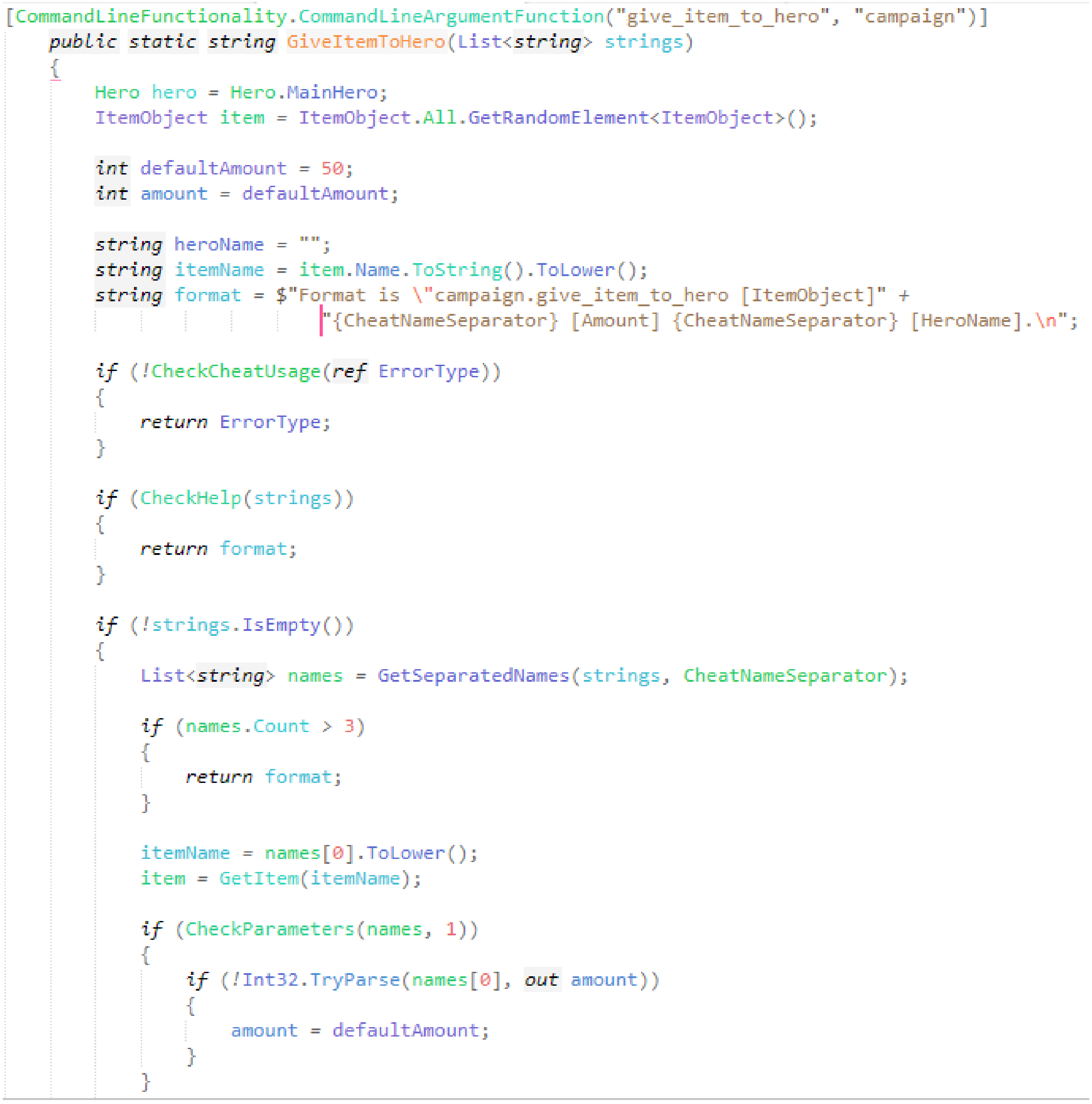
get_lords_inside_settlement	set_hero_age
give_xp_to_party	find_hero
give_item_to_hero	set_campaign_speed
give_item_to_main_party	set_all_skills_of_given_hero
add_development_progress	find_mobile_party
escape_from_party	



```
public static ItemObject GetItem(string itemName)
{
    foreach(var item in ItemObject.All)
    {
        if (string.Equals(item.Name.ToString(), itemName, StringComparison.OrdinalIgnoreCase))
        {
            return item;
        }
    }
    return null;
}
```

Figure 2.15.1: An Example of the New Functions Created

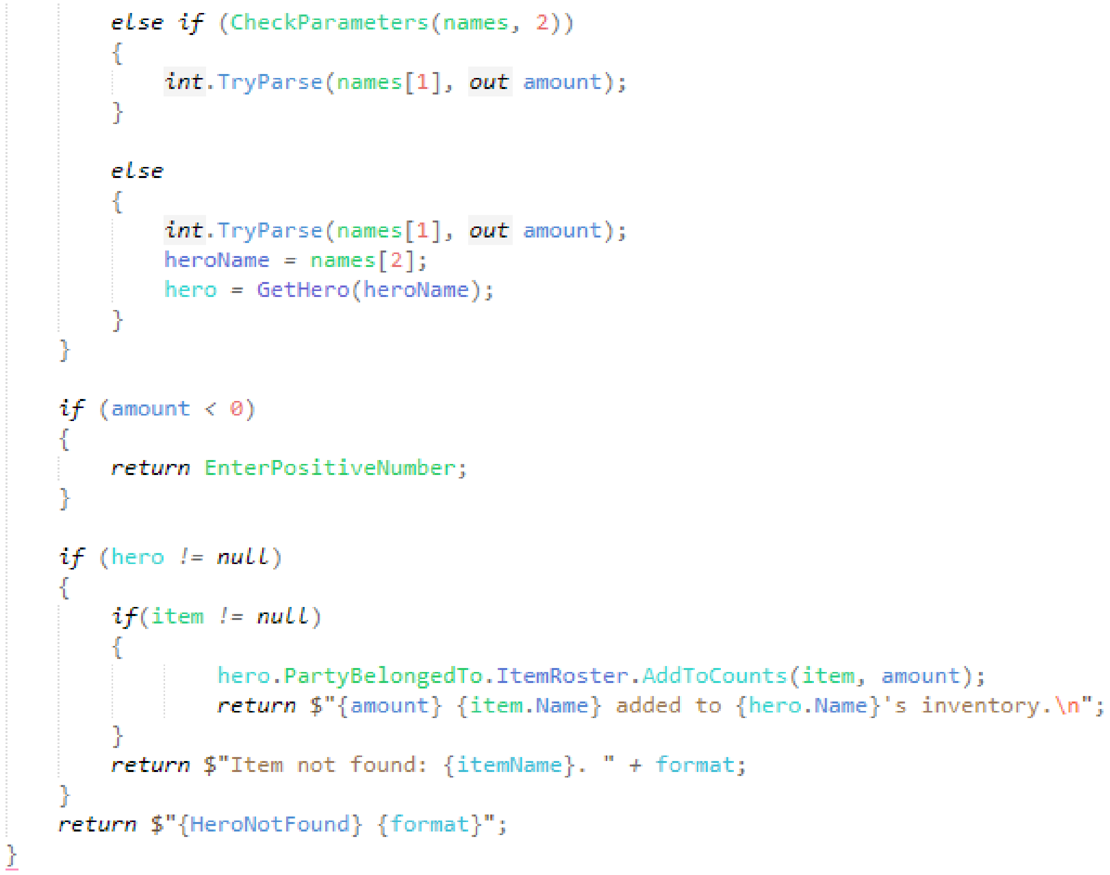
The example function above allows each item call to be written into the cheat, regardless of whether it is capital or small.



```
[CommandLineFunctionality.CommandLineArgumentFunction("give_item_to_hero", "campaign")]
    public static string GiveItemToHero(List<string> strings)
    {
        Hero hero = Hero.MainHero;
        ItemObject item = ItemObject.All.GetRandomElement<ItemObject>();

        int defaultAmount = 50;
        int amount = defaultAmount;

        string heroName = "";
        string itemName = item.Name.ToString().ToLower();
        string format = $"Format is \"campaign.give_item_to_hero [ItemObject]" +
                        "{CheatNameSeparator} [Amount] {CheatNameSeparator} [HeroName].\n";

        if (!CheckCheatUsage(ref ErrorType))
        {
            return ErrorType;
        }

        if (CheckHelp(strings))
        {
            return format;
        }

        if (!strings.IsEmpty())
        {
            List<string> names = GetSeparatedNames(strings, CheatNameSeparator);

            if (names.Count > 3)
            {
                return format;
            }

            itemName = names[0].ToLower();
            item = GetItem(itemName);

            if (CheckParameters(names, 1))
            {
                if (!Int32.TryParse(names[0], out amount))
                {
                    amount = defaultAmount;
                }
            }
```

Figure 2.15.2: An Example Source Code of Cheats Part 1

The cheat example given above provides the user with the item he/she chooses to the desired hero after typing *campaign.give_item_to_hero* on the debug screen. While doing this, the separator symbol ‘|’ must sign. If the player forgets any of them, a randomly selected item is given to the number of main heroes I have identified. If there is a misspelling in the key code, the player is informed to use the correct format.



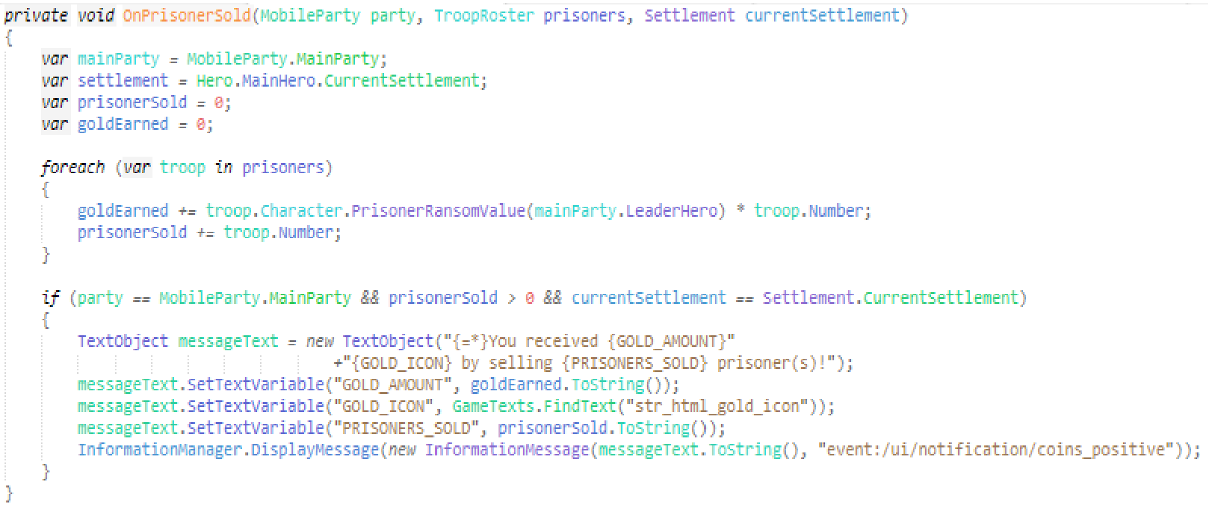
```
        else if (CheckParameters(names, 2))
        {
            int.TryParse(names[1], out amount);
        }

        else
        {
            int.TryParse(names[1], out amount);
            heroName = names[2];
            hero = GetHero(heroName);
        }
    }

    if (amount < 0)
    {
        return EnterPositiveNumber;
    }

    if (hero != null)
    {
        if(item != null)
        {
            hero.PartyBelongedTo.ItemRoster.AddToCounts(item, amount);
            return $"{amount} {item.Name} added to {hero.Name}'s inventory.\n";
        }
        return $"Item not found: {itemName}. " + format;
    }
    return $"{HeroNotFound} {format}";
}
```

Figure 2.15.3: An Example Source Code of Cheats Part 2

2.16. Add a New Notification to the Default Notifications

When the prisoners under the player's command are exchanged, the notification indicating the number of prisoners and the amount of money earned is added to the debug screen within the game.

As a result of the feedback I received from my supervisor while solving the problem, I made some corrections and sent them for re-evaluation. My supervisor said he wanted to make a big reorganization of the part I'm working on. So I had the opportunity to watch him closely while he was working. In this way, I have seen what I need to pay attention to during the thinking stages and more effective ways of researching relevant sections within the project. The problem was solved after getting the necessary arrangement from the person concerned for the UI portion of the problem.



```
private void OnPrisonerSold(MobileParty party, TroopRoster prisoners, Settlement currentSettlement)
{
    var mainParty = MobileParty.MainParty;
    var settlement = Hero.MainHero.CurrentSettlement;
    var prisonerSold = 0;
    var goldEarned = 0;

    foreach (var troop in prisoners)
    {
        goldEarned += troop.Character.PrisonerRansomValue(mainParty.LeaderHero) * troop.Number;
        prisonerSold += troop.Number;
    }

    if (party == MobileParty.MainParty && prisonerSold > 0 && currentSettlement == Settlement.CurrentSettlement)
    {
        TextObject messageText = new TextObject("{=*}You received {GOLD_AMOUNT}"
                                                +"{GOLD_ICON} by selling {PRISONERS_SOLD} prisoner(s)!");
        messageText.SetTextVariable("GOLD_AMOUNT", goldEarned.ToString());
        messageText.SetTextVariable("GOLD_ICON", GameTexts.FindText("str_html_gold_icon"));
        messageText.SetTextVariable("PRISONERS_SOLD", prisonerSold.ToString());
        InformationManager.DisplayMessage(new InformationMessage(messageText.ToString(), "event:/ui/notification/coins_positive"));
    }
}
```

Figure 2.16.1: New Notification Source Code

2.17. Recruiter Cultures In Conquered Cities

Soldiers produced in the seized cities were provided according to the culture of the new existing kingdom. The necessary conditions were updated and production is made in the old cultures.

2.18. About the Assigned but Rejected/Closed Issues

I have observed that some of the issues and tasks assigned by my supervisor have already been solved in different branches or are in the solution phase by other workers. Therefore, the related issues were closed or rejected by the notifier via Jira according to their concerns and my report does not include detailed information about these issues.

3. PROJECT ORGANIZATION SOFTWARES

3.1 Plastic SCM

Plastic SCM is a cross-platform commercial distributed version control tool developed by Códice Software Inc. It includes a command-line tool, native GUIs, diff and merge tool and integration with severalIDEs. It includes these components:

- Server
- Command-line client
- GUIs
- Diff and merge tools
- Web-based admin interface

This software is used to simplify error and work, tracking by allowing the work to be tested multiple times before adding it to the main part. It also makes it easier to recover by examining the saved branches and changesets of the project.

3.2 JIRA

JIRA is a product management tool that the company uses which includes:

- **Issue tracking & management:** More precise way to define workflow, manage daily task and track the progress of the work done by the team.
- **Product Management:** Where you can create the project, create sprints inside those projects, manage it by creating stories which team would work on to deliver the product.
- **Reporting:** Gives the best reports to understand the progress of the project. Helps you with total spent verses the estimation, which us to understand the budget very preciously.
- **Scrum & Kanban:** Helps you with configurable scrum and Kanban boards. Where you can use Scrum board for managing your board, which you can use Kanban to manage bugs, epics in the system.

4. CONCLUSION

I completed my 4 weeks internship at Taleworlds Entertainment. With the Campaign team, I contributed to the editing of some bugs on the gameplay, adding new features, and making the code more legible. To do this, I had to investigate the functions within the project and learn the correct C#, game engine and previously written functions to implement them appropriately. At the end of my internship, I became more knowledgeable about developing large-scale games.

Every day during my internship, I expressed all my work in the afternoon team meetings, and I also learned about the parts which they worked. Thanks to these meetings led by our team leader, I got an idea of the ways and challenges of multi-thinking on the project. Besides, I had a great chance to improve myself in business and human relationships. The professional working conditions of the company and the network of relationships made me a great contribution to this. Thanks to this communication line I have gained, I had an opportunity to meet the new people where I can get support when I need help or when I am in doubt about how to practice the information I have learned in my school life. I also believe that every practice I have obtained here will contribute to both my school life and my future studies.

Belatedly, I can say that the most significant benefit of my internship grants me some useful information about game development, which is a profession I would like to work on in the future. Moreover, I am very felicitous to be involved in the creation of a game like Mount and Blade II: Bannerlord, which has received a great deal of international recognition and to be able to benefit from the knowledge and experience of the people there.

5. BIBLIOGRAPHY

https://app.grammarly.com/

https://translate.google.com

https://en.wikipedia.org/wiki/Plastic_SCM

https://reviews.financesonline.com/p/jira/

https://docs.microsoft.com/tr-tr/dotnet/csharp/

https://www.taleworlds.com/

https://en.wikipedia.org/wiki/TaleWorlds_Entertainment