Project pitch

Group 71

Team

Name	Student number	Email
Håvard Skjærstein	2780502	h.skjaerstein@student.vu.nl
Kloena Burazeri	2723423	k.burazeri@student.vu.nl
Andrei Daraban	2727462	a.daraban@student.vu.nl
Lara Ichli	2730901	l.ichli@student.vu.nl

Overview

A virtual pet simulation game was popularized in the 1990s.

We will be designing and implementing a modern take on the classic game.
Your virtual tamagotchi is accessible through a command line interface but still provides some simple nostalgic graphics.

The overall idea of our game is that the user designs and takes care for their virtual character by meeting its needs, such as feeding it, playing with it, cleaning it and ensuring it gets enough sleep.

The tamagotchi has a life cycle where hours are equal to years and you should meet its needs as stated by your creature in the beginning of the game.

The creature has different levels of hunger, happiness, energy and cleanness out of 100, available to the user through status bars.
The user must attend to these needs within a certain time limit which is different for each need. If the user fails to meet the tamagotchis' needs, the game will end and the creature will die.

We are drawing inspiration from the original Tamagotchi game, which can be found at the following URL: https://www.youtube.com/watch?v=h3pMVCYM_EM&ab_channel=ilovemytamas

Functional features

ID	Short name	Description	Champion
F1	Virtualization	The game begins with virtualizing the tamagotchi. The user can choose between various characters (human, cat, bird, bunny) with different features and name their pet.	Lara
F2	Life cycle	The tamagotchi has a different life cycle in which 1 hour equals 1 tamagotchi year.	Håvard
F3	Bathroom/ Cleaning	The tamagotchi needs to be taken to the bathroom and be cleaned. If the tamagotchi is not taken care of by the user, the cleanness level of the tamagotchi decreases. (Details of the level bar indicating cleanness can be found in F7. How much the level decreases and when?)	Kloena
F4	Feeding	The tamagotchi sends a notification by printing a message on the screen when it is hungry. If it's not fed, the hunger levels decrease. (Details of the level bar indicating hunger and happiness can be found in F7.)	Andrei
F5	Entertainment	The tamagotchi requires the user to be happy with several mini games and activities. Mini Games: There are two mini games, guess the number and guess the character. If it takes more than 10 guesses for the user to find it, they lose the mini game and the happiness attribute loses 10 points.	Havard

Functional features

F6	Sleep & Wake	The tamagotchi notifies the user by printing a message on the screen when it needs to go to sleep and the user should take it to sleep otherwise the sleepiness level of the tamagotchi goes down. (Details of the level bar indicating energy can be found in F7.)	Lara
F7	Status	The user can observe the cleanness, hunger, happiness, energy levels of their pet through the status feature. They can also see the character and the name of their pet and id. For hunger and sleepiness level the bar starts at 0 and goes to 100, when it reaches 100 tamagotchi dies. For cleanness and happiness the bar starts from 100 and goes to 0. When it reaches 0, the tamagotchi dies. The tamagotchi needs to be: Fed in every 2 minutes Played with every 5 minutes Taken to sleep in every 7 minutes Cleaned every 3 minutes If the needs are not met in those times, the levels decrease by 10 or increase (hunger) by 10 indicated by hashtags in the status bar. Each time the user meets a need of their creature, level bar increases by 10.	All

Quality requirements

ID	Short name	Quality attribute	Description
QR1	User interface	Usability	The game should have a user-friendly interface that is easy to learn and navigate with command-line figures depicting the state of the creature and its vital values.
QR2	Stability	Reliability	The tamagotchi should behave the way it should when increasing/ decreasing its vital levels(hunger, cleanness, happiness, energy) according to the player's actions. It should also respond without crashing or any errors to the users' actions.
QR3	User Experience	Maintainability	The software should be easy to maintain and update, with clear documentation and modular design. If more features would like to be added to the game in the future, the code and the documentation should be unambiguous for the developers and stakeholders.
QR4	User Experience	Performance	The player should be able to perform their actions and get responses from their creature without any delays or errors in an interval of less than 3 seconds.

Time log

Assignment 1

Team Member	Activity	Week	Hours
Lara	Write the discussed functional features.	1	20 min
Havard	Write overview page of the pitch slides.	1	30 min
Andrei	Write the discussed quality requirements.	2	30 min
Kloe	Create the time log.	2	30 min
Havard	Create the contract.	2	20 min
Lara	Arrange the meetings for the contract.	2	10 min

Signed contract

The link of the contract can be found below:

https://docs.google.com/document/d/1dLklp7Cj6G4cAwqsQsQzBRnWLhrsOa9WGy0TAnPgCDw/edit?usp=sharing

Thank You for Listening!