A picture containing text, vector graphics, light

Description automatically generated

Table of Contents

**Creation of the Educational Traveling game** 3

Team information 3

Introduction 3

Ways of realization 4

**Work plan** 5

Added Features 5

**Creation of the Educational Traveling game**

**Team Information**

|  |
| --- |
| Team Roles |
| Valeri Ivanov – Scrum Trainer |
| Alexander Manov – Backend Developer |
| Galin Georgiev – Backend Developer |
| Petar Matsaliev– QA Engineer |

**Introduction**

|  |
| --- |
| Introduction |
| What is the product?  The product is our depiction of the “Educational Traveling” theme. We used C++, Eight-ball engine and Raylib. |
| How can you access it?  You can read about our collaborative work on GitHub and access our project’s repository files. |
| What about communication?  We communicated through Teams due to its helpful functions like screen sharing and text channels. The team was well connected and the work was efficient. |
| What programs were used?  We used GitHub for file management and collaborative work, Visual Studio 2022 for code editing, MS Teams for communication, MS PowerPoint for our presentation, MS Word and GitBook for the documentation, MS Excel for our QA documentation, Doxygen for our code documentation and Figma for our design. |

**Ways of Realization**

|  |
| --- |
| How it was done. |
| Distribution of tasks  The Tasks were distributed based on the capabilities of the teammates. If major tasks came up and they proved themselves to be difficult, then the team would gather to discuss the faced issue. This way our team was as productive as possible. |
| Task Completion  Every day the team held a meeting to track the development of the app and help each other progress further. We also resolved issues and shared ideas. |
| Deadlines  In meetings tasks and deadlines were distributed among the teammates. The team also kept track of how tasks were coming along and discussed issues and possible solutions to given issues and further discussions were made for improvement. |
| Branches  We decided to use branches so we could split our work individually and make production more efficient. The code, major code tasks, documentation and design were split into their own respectful branches |

WORK PLAN

**Added Features**

|  |
| --- |
| Further Feature Evaluation |
| Create the main menu  The main menu was created by our Backend developers. It is used to navigate through the application. |
| Create the character carousel  The character carousel was created by our backend developers. The design was made by our scrum trainer. The character carousel is used to choose between characters. |
| Map Renderer  A map which visualises the loaded GeoJSON from the map pack developed by our backend developers. |
| Status View  Shows all resources and winning probability along with the won/lost ratio in the left bottom corner. |
| Action Buttons  Action Button #1: Puts you in organisation mode so you can click on a country on the map to make an organisation on it.  Action Button #2: Makes a national revolution, if the calculations are over a certain winning threshold the revolution is successful.  Action Button #3: Leads to another menu which lets you buy different resources which make the chance of winning higher. |
| GitBook  A modern documentation platform where teams can document everything from products to internal knowledge bases |
| Create the Doxygen documentation  The Doxygen documentation was created by our QA Developer and Scrum trainer. You can see in great detail how the code works through this documentation. |
| Create the QA Documentation  The QA Documentation was created by our QA Engineer. There are unit test reports about the application. |
| Documentation  The Documentation was done by the QA Engineer and the Scrum Trainer. The program used to do to that was MS Word. |
| Presentation  The presentation was created by our Scrum trainer to explain the concept of the application. |