7/27/2022

LYNN, Tom (txlyn3)

Ferny Grove State high school

Digital Solutions

FIA4 - Folio

Table of Contents

[Part 1 - Research and Investigation](#_Toc144323389)

[Stimulus insert](#_Toc144323390)

[Recognise and describe key elements of your solution](#_Toc144323391)

[Any existing data exchange applications](#_Toc144323392)

[Components of these systems](#_Toc144323393)

[Security processes](#_Toc144323394)

[Prescribed criteria Data:](#_Toc144323395)

[Self determined criteria](#_Toc144323396)

[Symbolise using mind maps and one or more of constructed sketches, annotated diagrams, images or screenshots](#_Toc144323397)

[Explain](#_Toc144323398)

[Analyse](#_Toc144323399)

[Evaluate](#_Toc144323400)

[Determine data exchange system requirements](#_Toc144323401)

[Part 2 – Data Exchange Solution](#_Toc144323402)

[Symbolise using mind maps and one or more of constructed sketches, annotated diagrams, images or screenshots](#_Toc144323403)

[Explain](#_Toc144323404)

[Synthesise](#_Toc144323405)

[Evaluate](#_Toc144323406)

[Digital data exchange solution against determine prescribed and self-determined criteria](#_Toc144323407)

[Development Log](#_Toc144323408)

[Data dictionary](#_Toc144323409)

[Programming with comments](#_Toc144323410)

[Part 3 – Impacts](#_Toc144323411)

[Recognise and describe key elements](#_Toc144323412)

[Risks associated with storing and accessing data](#_Toc144323413)

[digital security strategies, including authentication and encryption strategies](#_Toc144323414)

[Analyse a data security problem to identify risks](#_Toc144323415)

[Determine a security strategy for data](#_Toc144323416)

[Evaluate against prescribed and self-determined criteria the impact of data transmission on personal, social and economic needs (written individually)](#_Toc144323417)

[Recommend an appropriate strategy to increase data security. (written individually)](#_Toc144323418)

# Part 1 - Research and Investigation

## Stimulus insert

**Technical specifications**

**Identification**

Governments have developed strategies aimed at encouraging young people to achieve their gaming potential. The Australian Institute of gaming (AIG) heavily support this agenda, as a result they have created a new competition to promote youth game development. A proof of concept is required to demonstrate the planning and data transfer functionality of a new application developed by young people.

A new application developed by young developers should:

• use API datasets (weather API for example)

• allow for users to retrieve and update profile information

• Explore the use of non-player characters

The proof of concept involves:

• developing a low-fidelity prototype of the application

• generating the data exchange component that simulates exchange of data between two digital systems (data server and application). The solution will receive data in one format and programmatically transform it into another format for displaying

• evaluating impacts and making recommendations for improving data security during transfer

• developing a video to demonstrate data transfer functionality

**Interactions**

Proto-personas have been developed for potential users of the application (see Figure 1 below).

**Figure 1: Sample user profiles for the new UNITY application**

|  |  |
| --- | --- |
| Joe Biden - Wikipedia | **Joe “Sleepy” Mamah**   * Hasn’t had a promotion since 2020 * Falls asleep a lot * Avid gamer * Large vocabulary and never stumbles over his words |
| Is Scott Morrison in Hawaii? A very serious investigation. | **Scotty Morris**   * Love to go on vacation and play video games whilst ignoring problems * Love tackle sports – this is reflected by his love of video games * Coal Enthusiast * Excellent relationships with the French |
|  | **Ollie RM**   * Intense gamer, with a passion for puzzle games * Wants to become a game developer later in life, he also has passions for software engineering. * He loves 3d games, with a particular interest in open world ``exploration. * Wants to improve in his orienteering, and communication skills |

**Component specifications**

**Data**

The new application must:

• incorporate the use of data to track player info

The UNITY client must:

• connect to each dataset

• receive data in one format and programmatically transform it into another format and display the data

• display the contents of each dataset in a clear and useful format

* User statistics, last save point, etc.

• include relevant headings for each column of data

• not directly display information taken from weather API, in order to add realisms to the game, excepting situations wherein NPC’s discuss the weather.

**User interface/experience**

The new application must:

• be accessible on any of the chosen output devices (UNITY builds)

• allow the user view\update profile data

• feature a responsive interface.

**Code**

The new application must include:

• an algorithm to retrieve and display the data from each API

• code to retrieve data from the API

• code accuracy

- data is to be displayed correctly

- the client needs to

use datasets

connect to normalized dataset

display only the stated data

include column headings for the data

 code efficiency.

## Recognise and describe key elements of your solution

### Any existing data exchange applications

**Muck** is a Indie survival multiplayer game that was developed by Dani, and is a great example of example of semi-open world games. This aspect of muck is one that will be drawn and improved upon in the generated solution. Another aspect of this game that the generated solution is the low poly style of the game. The appeal of low poly art can be attributed to several factors. One reason is that it evokes a sense of nostalgia for early video games, which often used low poly graphics due to hardware limitations. Another   
reason is that it can be used to create a unique aesthetic that stands out from other styles. Additionally, low poly art can be computationally cheap to render, making it an attractive option for indie game developers who may not have access to high-end hardware. For the reasons previously outlined, a low poly style will be used.

(MuckWiki, n.d.)

**The** **Talos Principle** is a first-person puzzle game developed by Croteam and published by Devolver Digital. The game is set in a world full of puzzles that require you to use logic and reasoning to solve them. The game has over 120 immersive puzzles in a stunning world. The game’s story is about humanity, technology, and civilization. You can uncover clues, devise theories, and make up your own mind. The game has won several awards for its innovative gameplay, thus one of the aspects of this game that will be used is the puzzle-driven narrative of the game. The puzzles that will be implemented are not just there for the sake of being puzzles, but they are an integral part of the story. The puzzles will be designed to make you think about the story and the world around you.

(PC Gamer, n.d.)

**FIFA 21**, the widely acclaimed football simulation developed by EA Sports, has heralded a paradigm shift in the realm of gaming. Beyond its visually stunning graphics and deeply immersive gameplay, FIFA 21's distinctiveness lies in its judicious utilization of Application Programming Interfaces (APIs). These digital conduits proficiently bridge the virtual landscape of the game with the dynamic real-world football ecosystem, facilitating the seamless dissemination of real-  
time data encompassing player statistics, team particulars, and live match scores, thereby profoundly enhancing the gaming experience.

(Mehta, 2022)

In our innovative Endeavor, we endeavour to harness this API-driven ingenuity for a purpose that is divergent yet equally compelling. Instead of emulating FIFA 21's football-centric API paradigm, we venture into uncharted territory with a focus on intellectual engagement. In our pioneering solution, we introduce a radical departure from the norm by replacing football-related data with an equally riveting subject matter: riddles. In this revolutionary paradigm, a dedicated riddles API will assume prominence. It will curate an ever-evolving compendium of cerebral conundrums, enigmas, and puzzles, thereby transmuting the user experience. Our innovation transcends mere entertainment; it is a catalyst for intellectual stimulation, nurturing curiosity and honing problem-solving acumen. In this manner, our solution, much akin to FIFA 21's transformative influence in the gaming arena, stands as a groundbreaking innovation, redefining the contours of intellectual engagement. Much like FIFA 21's API integration heightens the gaming experience, our avant-garde riddles API promises to introduce a novel and intellectually enriching dimension to our solution.

### Components of these systems

**From "Muck," the Indie Survival Multiplayer Game:**

* **Semi-Open World Gameplay:**
  + Inclusion: The semi-open world aspect from Muck will be adopted and improved upon.
  + Reason: This feature enhances player exploration and freedom within the game, providing a sense of vastness and discovery that can be captivating.
* **Low Poly Art Style:**
  + Inclusion: The low poly art style of Muck will be used.
  + Reason: Low poly art offers an aesthetically unique and computationally efficient visual style, making it appealing for indie game development. It can also evoke nostalgia and stand out from more realistic graphics, which can enhance the game's identity.

**From "The Talos Principle," the First-Person Puzzle Game:**

* **Puzzle-Driven Narrative:**
  + Inclusion: The puzzle-driven narrative concept from The Talos Principle will be integrated.
  + Reason: This approach ensures that puzzles are not just isolated challenges but integral parts of the story and world. It engages players in critical thinking, immersion, and a deeper understanding of the game's narrative.

**From "FIFA 21," the Football Simulation Game:**

* **API Integration:**
  + Inclusion: Inspired by FIFA 21's use of APIs, we'll incorporate a similar API-driven mechanism.
  + Reason: APIs enable real-time data and interactivity, enhancing the gaming experience. In our case, we'll use a riddles API to engage users intellectually, fostering curiosity and problem-solving skills.
* **Innovation and Uniqueness:**
  + Inclusion: Like FIFA 21, the generated solution aims to innovate and offer a unique gaming experience.
  + Reason: Drawing inspiration from FIFA 21 ensures that our solution incorporates cutting-edge features and interactivity, making it stand out in the gaming landscape.

In summary, the generated solution will amalgamate the best aspects of Muck's semi-open world gameplay and low poly art style, The Talos Principle's puzzle-driven narrative, and take inspiration from FIFA 21 by incorporating API integration for real-time engagement and innovation. This synthesis aims to provide players with a distinctive, intellectually stimulating, and immersive gaming experience that combines exploration, puzzle-solving, and unique aesthetics while embracing the innovations seen in FIFA 21's API integration.

### Security processes

|  |  |
| --- | --- |
| **Encryption**:  Implementation: Encrypt sensitive user data, such as passwords, before storing it in the database using strong encryption algorithms like bcrypt or Argon2.  Reason: Encryption ensures that even if the database is compromised, the stored data remains unreadable to unauthorized users, enhancing data security. | **Password Hashing:**  Implementation: Hash user passwords using strong, one-way hashing algorithms like SHA-256 or bcrypt.  Reason: Password hashing ensures that even if the database is breached, attackers cannot easily reverse-engineer passwords from the stored hashes, enhancing user password security. |
| **Access Control:**  Implementation: Implement role-based access control (RBAC) to restrict access to sensitive user information to authorized individuals only.  Reason: Access control ensures that only users with specific roles or permissions can access certain parts of the system, reducing the risk of unauthorized data access or manipulation. | **Two-Factor Authentication (2FA):**  Implementation: Enable 2FA as an optional security feature for user accounts, requiring users to provide a second authentication factor, such as a one-time code from a mobile app, in addition to their password.  Reason: 2FA adds an extra layer of security, making it significantly more difficult for unauthorized users to gain access to accounts, even if they have the password. |

These data security methods collectively strengthen the user login system's defences, safeguarding user data and enhancing overall system security.

## Prescribed criteria Data:

* Incorporate the use of data to track player information

UNITY client:

* Connect to a database and pass information to and from the database

Code:

* Code to retrieve data from the API.
* Code accuracy: data is to be displayed correctly.

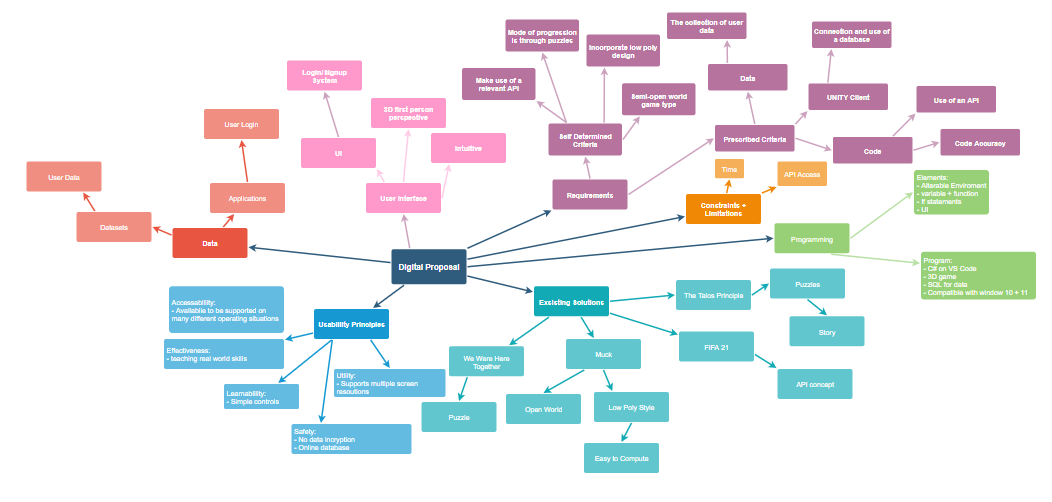
Usability:

* Adhere to the usability principles

## Self determined criteria

* Mode of progression is through puzzles
* Semi-open world game type
* Incorporate low poly design
* Make use of a relevant API

## Symbolise using mind maps and one or more of constructed sketches, annotated diagrams, images or screenshots



A diagram of a computer flowchart

Description automatically generated

|  |  |  |
| --- | --- | --- |
| Explain | Analyse | Evaluate |
| - data interface, data structures and data specifications  - digital methods of exchanging data | **- data exchange problem to identify**  **- the data structures, including data input and output requirements**  **- data exchange methods** | **evaluate against prescribed and self-determined criteria the most suitable process for exporting and importing data between the two digital systems.** |
| The game will make use of a local server to connect to a database, this will then allow the unity client to make an HTTP web request to point to a PHP file, and pass data that the user enters on the client side, through the POST method to the database. The structure of this database will be clear and consistent storing player information in tables with clear table headings and datatypes. This will allow for the game to have an account creation and login system in order for the players. The second data exchange method that will be used is the use of an API to request information relevant to the game. | The data exchange problem is the storage of user data, so that when users come back to the game after closing it, their information will be stored. The data input will come from the user through unity UI input field when users create accounts, and login to that account. Structure will make use of standard user collection data types such as userId, email, password etc. finally the exchange method that will be used are the post method. | The use of these fits the prescribed and self determined criteria in that as per the criteria, the unity client will make web requests to connect to a local server, thus aligning with prescribed criteria of unity making db connection, and incorporating the use of player information tracking. The second data exchange method of the API, aligns with the self determined criteria of the relevant API in the use of the riddles API. |

|  |  |  |
| --- | --- | --- |
| Determine data exchange system requirements | | |
| Db Connection   What is REST API | PHPenthusiast  The POST method is one of the HTTP methods used to send data to a server to create or update a resource. When you use the POST method, you send data in the body of the request, which is not visible in the URL. This makes it more secure than GET requests because sensitive information is not exposed in the URL. The POST method is typically used when you want to create a new resource on the server or update an existing one. It’s also useful when you want to send large amounts of data because there are no limits on the amount of data that can be sent in the body of the request | We chose to use the POST method to send data to a server to create or update a resource. When you use the POST method, you send data in the body of the request, which is not visible in the URL. This makes it more secure than GET requests because sensitive information is not exposed in the URL. The POST method is typically used when you want to create a new resource on the server or update an existing one. It’s also useful when you want to send large amounts of data because there are no limits on the amount of data that can be sent in the body of the request. | **Effectiveness**: The use of the POST method for both the Unity web request and the API web request is effective in achieving their intended purposes. The POST method is commonly used to send data to a server to create or update a resource, which is exactly what these requests are doing.  **Utility**: The use of the POST method for both requests provides value to users by allowing them to create or update resources on the server. This is an important feature for many applications.  **Learnability**: The use of the POST method may not be immediately intuitive for all users, but it is a well-established convention in web development. Users who are familiar with web development will likely be familiar with the POST method.  **Safety**: The use of the POST method is generally safe, as it allows sensitive information to be sent in the body of the request rather than being exposed in the URL. However, it is important to ensure that appropriate security measures are in place to protect user data and prevent unauthorized access.  **Accessibility**: The use of the POST method does not have any significant impact on accessibility, as it does not affect the ability of users with disabilities to access and use digital solutions.  Overall, the use of the POST method for both requests is generally consistent with usability principles. However, it is important to ensure that appropriate security measures are in place to protect user data and prevent unauthorized access. |

# Part 2 – Data Exchange Solution

## Symbolise using mind maps and one or more of constructed sketches, annotated diagrams, images or screenshots

A diagram of a computer

Description automatically generatedA diagram of a software process

Description automatically generated

|  |  |  |
| --- | --- | --- |
| Explain | Synthesise | Evaluate |
| the generated solution has sever methods of data transportation between systems, the primary method used for both the DB queries and the API used is the post method, and as per [data exchange requirements](#_Determine_data_exchange). The coded components control the flow of data securely through a password locked database which is accessed through PHP, which are pointed to through a unity web rtequest which is done through the post methods which is considered secure. Some examples of this code is below:  //create account in unity function() createAccount -> {  webrequest (php File, emailInput, passwordInput, passwordConfirm)  }  // php file  Connect to db  If passwordInput = passwordConfirm {  Db.run (“Select from userdata where email = emailInput”  If num rows > 0 {account already exists } else {  Db.run(“insert into user data values (email, password)  }  }  // api call  CallApi(){  Webrequest (apiUrl)  Result = sendwebrequest  Riddle = result.riddle  Answer = result.answer  Booktext.text = riddle  } | As seen in the pseudocode, the coded versions of these scripts have been coded and can be seen in the programming with comments section. This has four scripts that relate to both the prescribed criteria and relate to the protopersona of ollie rm. One of the coded components shown is the controller for the mini map in order to assist this persona wit4h their orienteering skills. The second coded component is the coded input manager that uses unity’s new Input system and utilises action maps to allow the controlling of the player. The final two scripts that can be seen are all related to data handling. The first script relates to the first and second portion of pseudocode, this handles the user account creation in the c#, which points to a php file, which connects to the database and confirms the user details, then if the account is available, creates the account. The next data related coded component is the api call, which uses the random riddles API. | After testing which can be seen recorded below in the development log, the account creation had some issues. Specifically the issue was the verification of email address, wherein the php script was set to select all from the table, where the email field, is equal to the users inputted email. If the query returned more than 0 rows, this meant that an account with that email already exists, else if there was 0 rows, it would insert the user data into the db. However upon testing and implementation of this code, it would fail to create a new account. This remains a bug that has not been fixed.  These data exchange solution align to the prescribed criteria of  - Connect to a database and pass information to and from the database to Incorporate the use of data to track player information  - Code to retrieve data from the API.  It achieves this as unity creates the connection to the database through using a web request through a local server to point to a PHP script with a transmitted form with user info they submitted.  As well the code uses a unity web request to call the API and utilise said JSON from the API in the game space.  The usability of these coded components align with the usability principles of safety, effectiveness, learnability, utility, and accessibility. The safety of the data transportation is currently at a base level through using the post method, however further encryption should be implemented. the data exchange solutions are effective as the code is simple, and not overcomplicated and do not handle large sums of data. The UI for the account creation follows the universal account creation template of username, email, password, and password confirm, therefore the learnability is very good. The utility of theses exchange systems are that it allows for the game to store player information to make tracking progress far easier. Finally the accessibility of this is limited, however it does allow for the use of colour blind people in terms of data exchange, as well as deaf people due to the use of black and white colour for the account page, and the use of the game space implementation of the data, specifically the riddle API not relying on sound. |

|  |  |
| --- | --- |
| Digital data exchange solution against determine prescribed and self-determined criteria | |
| Prescribed   * Connect to a database and pass information to and from the database to Incorporate the use of data to track player information * Code to retrieve data from the API. * Code accuracy: data is to be displayed correctly. * Adhere to the usability principles | **Self-determined**   * Mode of progression is through puzzles * Semi-open world game type * Incorporate low poly design * Make use of a relevant API |

|  |  |  |  |
| --- | --- | --- | --- |
| Development Log Make refinements and justified recommendations for current and future improvements. | | | |
| Date | **Type**  **(Feature/Test/Modification)** | **Description** | **Status** |
| Aug 4 2023 | Basic Player Movement and camera movement | Created scripts for player controller and camera controller | Was buggy, should be updated |
| Aug 19 2023 | Updated player controller | Updated player controller using action maps instead of input manager (IM) as IM is outdated. | Completed |
| Aug 19 2023 | Base terrain added | Terrain added to the main scene using unity terrain systems, trees painted etc | Completed |
| Aug 20 2023 | Map added, and interactable scripts made or obelisk | Mini Map added for main scene, as well as integration of obelisks to travel between scenes | Completed |
| Aug 22 – 23 2023 | Puzzle 1 created | Puzzle one scene, and scripts created, as well as sound effects added for ambiance | Completed |
| Aug 28 2023 | Database creation | Creation of c# and php scripts for account creation | Not working – Email authentication not working |
| Aug 29 2023 | Db testing | Testing the database account creation | Not working - same reason as before |
| Aug 29 | Scene traveling scripts | Scripts to enable player to travel between scenes | In progress |
| Aug 29 2023 | Puzzle 2 Creation | Puzzle two creation, this puzzle uses the API to generate a riddle, API call is working, just need to get Riddle being put into the game space. | In progress |

## Data dictionary

|  |
| --- |
|  |
| **Column** | **Type** | **Null** | **Default** | **Comments** |
| PlayerId *(Primary)* | int(11) | No |  |  |
| PlayerEmail | text | No |  |  |
| Username | text | No |  |  |
| Password | text | No |  |  |

## Programming with comments

A screen shot of a computer program

Description automatically generated

# Part 3 – Impacts

## Recognise and describe key elements

### Risks associated with storing and accessing data

Storing and accessing data entails various risks. Chief among them is the threat of data breaches, which can lead to financial losses and reputational damage. Data corruption or loss due to hardware failures and the risk of unauthorized access through cyberattacks or insider threats are also significant concerns. Compliance with data protection regulations and the challenge of data obsolescence further complicate matters. To mitigate these risks, proactive cybersecurity measures, diligent data management, and an understanding of evolving threats are essential. Hence why there are methods that are employed to ensure data security.

### digital security strategies, including authentication and encryption strategies

**Hashing Passwords:**

One fundamental aspect of digital security is the practice of hashing passwords. Hashing involves converting a user's password into a unique, irreversible string of characters. This process adds a crucial layer of protection by ensuring that even if a database is compromised, the actual passwords remain hidden. When a user logs in, the system hashes the entered password and compares it to the stored hash value. If they match, access is granted. However, the strength of this strategy depends on the complexity of the hashing algorithm and the use of salt, a random value added to each password before hashing. This combination makes it significantly more challenging for attackers to reverse-engineer passwords from the hashes, enhancing overall security. Employing robust hashing techniques is a fundamental component of any secure authentication system.

**Encryption:**

Encryption is another cornerstone of digital security. It involves encoding data in a way that only authorized parties can decipher it. This process ensures that even if unauthorized individuals gain access to the data, they cannot make sense of it without the encryption key. In transit, data encryption protects information as it moves between devices or over networks, safeguarding it from interception by malicious actors. Similarly, at rest, encryption secures data stored on devices or servers. Robust encryption algorithms like AES (Advanced Encryption Standard) play a pivotal role in securing sensitive information across a range of digital platforms, from messaging apps to online banking. Effective encryption strategies are vital to maintaining the confidentiality and integrity of data in today's interconnected world.

**Authentication**

Authentication strategies are crucial in the realm of digital security. They serve as the initial line of defence against unauthorized access to systems and data. Authentication involves verifying the identity of users or devices attempting to access a network or application. Traditionally, this was achieved through simple username and password combinations. However, as cyber threats have become more sophisticated, multifactor authentication (MFA) has gained prominence. MFA requires users to provide at least two forms of verification, such as something they know (password), something they have (a smartphone or security token), or something they are (biometrics like fingerprints or facial recognition). This layered approach significantly enhances security, as it becomes exponentially more challenging for malicious actors to breach an account or system without the necessary authentication factors. As digital ecosystems continue to evolve, robust authentication strategies remain pivotal in protecting sensitive information and ensuring the integrity of digital interactions.

## Analyse a data security problem to identify risks

Storing raw user data presents a range of critical risks. From a system perspective, improper data storage practices can lead to vulnerabilities, such as unencrypted storage or inadequate access controls, potentially allowing unauthorized parties to access sensitive information. In terms of data security and privacy, the risk of data breaches looms large, exposing personal details and creating substantial financial and reputational consequences. Furthermore, there's the risk of data misuse, as poorly managed data can inadvertently or intentionally be used for malicious purposes, violating user trust and privacy expectations.

## Determine a security strategy for data

We have made the strategic decision to enhance our data security measures by implementing basic encryption and password hashing techniques. This proactive step underscores our commitment to safeguarding sensitive information, ensuring that even in the event of unauthorized access, user data remains protected. By implementing these fundamental security practices, we aim to fortify our defences against potential threats, bolster user privacy, and maintain compliance with data protection regulations.

## Evaluate against prescribed and self-determined criteria the impact of data transmission on personal, social and economic needs (written individually)

**Connecting to a Database and Player Tracking (Prescribed Criterion**): We have successfully connected to a database to store and track player information, this was done through the use of a local server, and using unity’s web request system to point to a PHP file that sits on the sever.

**Retrieving Data from the API (Prescribed Criterion)**: We have implemented the necessary code to retrieve riddles from the API seamlessly. This ensures that players consistently encounter fresh and challenging content, enhancing their engagement and satisfaction.

**Code Accuracy (Prescribed Criterion)**: Our code is meticulously designed to display riddles accurately, ensuring that players receive the intended challenges. This precision in data presentation is crucial to maintain the game's credibility and player trust.

**Adherence to Usability Principles (Prescribed Criterion**): We have placed a strong emphasis on usability principles to make the game user-friendly, accessible, and intuitive. This commitment enhances the personal gaming experience and promotes social inclusivity by ensuring that players of all backgrounds can enjoy the game.

**Mode of Progression through Puzzles (Self-Determined Criterion):** The integration of the riddles API into the game's puzzle progression fulfills the self-determined criterion. As players solve riddles to advance, we cater to those who enjoy challenging and mentally stimulating gameplay.

**Semi-Open World Game Type (Self-Determined Criterion):** Our choice of a semi-open world game type encourages social interaction within the game world. Players can explore, collaborate, and share their riddle-solving experiences, aligning with the desire for socially immersive gaming experiences.

**Low Poly Design (Self-Determined Criterion):** We have successfully incorporated a low poly design into the game. This not only optimizes data transmission due to smaller assets but also caters to economic needs by reducing resource demands and potentially widening the game's audience reach.

**Relevant API Integration (Self-Determined Criterion):** The integration of the riddles API directly aligns with economic needs. It introduces the potential for monetization, perhaps through in-game advertisements or premium riddle packs, thus contributing to the game's economic viability.

In summary, by integrating the riddles API into the second puzzle of the game, we have not only met the prescribed and self-determined criteria but have also enhanced the overall gaming experience, personalized player progression, encouraged social interaction, optimized data transmission, and created potential economic opportunities, making this integration a strategic and well-rounded choice for our game.

**Personal:** the generated solution meets the target audience which can be seen in the protopersona Olliver RM, as he is an avid gamer that likes puzzle driven narratives, our innovative digital solution is a perfect fit. Combined with the use of the mini map, and location driven start to each puzzle, this perfectly aligns with Oliver’s goal to increase his orienteering skills, thus adding a component of utility to the usability of the solution.

**Social:** As this game is a single player adventure puzzle game, there is not a large social impact of this game, therefore it is recommended that the solution undergo further development to implement multiplayer functionality

**Economical:** It was found that the use of a low poly texture type, allows for the game to be run more economically, thus increasing the accessibility of the game for those with less capable devices.

Whilst the current state of the generated solution does meet the prescribed criteria, there is massive room for improvement.

## Recommend an appropriate strategy to increase data security. (written individually)

Given that due to time constraints, certain aspects of the proposed solution were not able to be generated into the final generated solution, such as the previously described data security methods, it is recommended that in order to increase the security of the data used and collected, that password hashing, and basic encryption methods like those previously outlined should be implemented. Implementing this would in turn greatly reduce the risks associated with user data collection.

References

Mehta, R. (2022, October 7). *FIFA 21 Team Management – How You Should Set Up Your Team Tactics to Dominate Your Opponen*. Retrieved from Fifa infinity: https://www.fifa-infinity.com/fifa-21/fifa-21-team-management-how-you-should-set-up-your-team-tactics-to-dominate-your-opponent/

MuckWiki. (n.d.). *Muck*. Retrieved from Wiki Fandom: https://muck.fandom.com/wiki/Version\_1.3

PC Gamer. (n.d.). *the talos principle*. Retrieved from pc gamer: https://cdn.mos.cms.futurecdn.net/69be1b3a4efd7fb99b49affd78007f11.jpg