**1. Introduction**The content of this section should explain the following:

**What is the problem you are trying to solve?**

Our project should create a platform where game developers can share their early-stage games for testing/ feedback gathering. Players would have a chance to familiarize themselves and test games before game release. Platform also will provide summary about the game based on players feedback by gathering them and summarizing via AI in a short article. That helps players to have short overview and also game developers get summary of feedbacks.

Guo: As an additional note to the above, this web application will primarily serve independent game developers or small indie game studios. It aims to integrate scattered indie game resources while providing a platform for developers with limited budgets and scales to connect with potential players during the game development process. Independent game developers will be able to create a homepage for their games in development. On this page, developers can post descriptions of the game, patch notes, playtest events, trailers, and more. Developers with needs can upload stage-specific game test builds for player testing, and quickly receive ratings and feedback from interested players about that stage's results, thus improving the final product more effectively. Players can browse and track the development progress of games they are interested in, participate in testing, and directly contact developers through the platform to leave suggestions that help improve the game.

**Why is this an important problem?**

Small indie game studious do not have place where they can easily test and get feedback about the game from players on the pre-release stage.

Guo: Although some well-known game distribution platforms like Steam and GOG do provide platforms for the release and promotion of indie games, these platforms focus more on promoting completed works or stage-complete early-access products and facilitating community interaction. Due to the large number of commercial rockstar AAA games on these platforms, the attention and support for indie games are often insufficient. Independent developers or studios with limited scale and budget need to start increasing their exposure early in the game development process to build a potential player base or to receive feedback and assistance from external sources. Currently, their options are mainly limited to setting up their own blogs or posting updates on personal social media platforms. Therefore, the platform we plan to develop aims to integrate these resources for indie games in development, allowing potential players to access information they care about in one place and helping developers address issues encountered during development, so they are no longer isolated and unsupported.

**Why is this a challenging problem?**

Because there is a significant barrier for small game studious to develop the game. They don’t have excise resources beside developing a game to communicate with players and gather their feedback.

Guo: For our platform, the challenges we foresee include how to manage the storage of test build packages and the feasibility and implementation of AI-generated analysis reports.

**What will you be developing to address the problem?**

The idea to develop a website that game studious and players can use for gathering feedback and observe pre-released games.

**Who are the intended users of your product, and how will the product benefit them?**

Game studious: benefit to have a place where they will get a feedback about the game on pre-released stage

Players: benefit to have a short game observation, can leave a feedback, can play in pre-released games

Guo:

Players:

to track the development progress of games they follow, participate in ratings and reviews, receive invitations to test, receive digital gifts from developers, and engage in crowdfunding activities.

Independent developers and indie game studio:

will be able to upload test builds, promotional videos, event notifications, digital materials (such as character designs and soundtracks), collect feedback on the current version, receive AI-generated analysis reports, and initiate crowdfunding campaigns.

**2. Related Work  
  
What are the existing approaches to solving this problem?**

Current approach for game developers to send direct personal invitation to the players to gather their feedback, analyze it, and incorporate changes after. Or skipping the part with public tests and release game without it.

Players familiarize with games after release without any objective information about the game or should read many comments to make up their opinion.

Guo:

itch.io: It is the website that specializes in indie game distribution.

Steam community: The world’s largest gamer community which has the indie game distribution business.

SNS accounts: Many game developers choose to use official social media account to promote their games like on Instagram/Discord.

**How will your approach be different/better?**

We going to create a place where small games studious and players can meet each other and share their projects, feedbacks and get short overview about the pre-released games.

Guo: For players, our platform will collect the pre-released indie game info and provide spaces for players to interact with developers and contribute to the WIP project.

For developers: Our platform will focus exclusively on indie games, providing a one-stop service for developers to upload game test builds and interact with potential players.   
  
**3. Approach  
  
How will you solve the problem?**

We going to create website. To give possibility to Studious register an account, publish their materials, games news, announcements. Players can register their accounts, leave a comment about the games on the website. AI will collect comments and create short summary for players and game studious.

**What product will you be building?**

The Website as a front-end interface and database (game studious and players accounts). Additionally, we will use AI API to summarize feedback for players and game studious.

**What is the software architecture?**

 Guo：We currently plan to adopt an architecture similar to MVC for building our platform.

**What are the algorithms involved?**

Create constructors for new accounts, comments, posts. Sort, store, transfer data from website to database and back.

Guo: it might be involving some search algorithms like searching based on keywords and filter search results on given fields like game genre or developers.

**How will you be implementing it?**

We will create the Website for the game studious and players. Gather their information in database and publish it on the website.

**What hardware will you use and why?**

We will use only cloud services for this project since we don’t have any common infrastructure to develop and deploy our project.

**What operating system will you use and why?**

It is web development therefore website should be OS neutral.

**What programming language will you use and why?**

Need to choose what we going to use for website building Python (Django and Flask) or JavaScript (React) – it is the most popular programming languages in web development

Database: MySQL or SQLite ---- SQL is the standard for database management

Guo: For the framework, currently we are evaluating between Django and ASP.net

**What libraries will you use and why?**

Depends from the language and database we choose.

**4. Deliverables  
  
State the deliverables (documents, features implemented in code, etc.) for EACH week.**

We going to prepare various diagrams (use case, class, state, sequence diagrams) and database architecture (tables) to split the particular tasks during development (ensure that we are on the one page in terms of necessary steps to be done)

**5. Evaluation  
  
How will you define success? E.g., accuracy, speed, ease of use, user adoption, user ratings.**

We will use auto-test for the website to test functionality and also user feedback for usability.

Guo：Core functionalities need to be implemented without errors; the user interface should be simple and aesthetically pleasing, making it easy for new users to get started and for experienced users to navigate quickly.

**How will you measure success? E.g., case studies, benchmarking.**

Firstly, all features should work without errors. Secondly, website should be intuitive and user friendly.

Guo：Before syncing each version to the main branch on GitHub, we will conduct local testing to ensure that the core functionalities run smoothly and to define and test various exceptions. After syncing, we will perform additional manual testing to ensure the stability of the current version.  
  
**1. Game Browsing and Discovery**

**Benchmark 1: Game Search Functionality**

* **Metric**: Game search results should be relevant, accurate, timely.

**Benchmark 1.2: Game Filtering and Sorting**

* **Metric**: Filters should be applied instantly, and sorting should correctly rank games by criteria such as popularity, release date, or rating.

2. Game Uploading for Developers

**Benchmark 2.1: Game Upload Time**

* **Metric**: Games should be uploaded within a reasonable time frame

**Benchmark 2.2: Game Metadata Submission**

* **Metric**: Developers should be able to upload metadata (game title, description, screenshots, etc.) within 1 minute without errors.

**6. Risks**

**What can go wrong?**

Volume of work might exceed our capabilities (time, skills, experience)

Guo: switch to Django might cause high learning curve because we don’t have much experience with this framework before; The storage solution for the website(for developers to upload videos and compressed files) might also be a potential issue because we don’t have experience on such large scaled storage solutions previously and it might requires paid cloud storage plan.

**What will you do to minimize the chances of things going wrong?**

We invest time in documentation aspect. To have clear plan what we need to do and how we going to implement it. It should give us clear pace to complete the project.

Guo: We are still evaluating the framework we will choose. If we decide on Django, we will first create a quick prototype to validate the feasibility of the process. Regarding the storage solution, we might not to handle storage on our own website. Instead, we may require developers to upload their own cloud storage links and video platform links.

**What will you do if things still go wrong?**

We going to have regular and close interaction during the full length of the project. (regular meetings and on-hoc calls)

Guo: and we will have alternative plans which might be easier on technique aspect such as the storage plan above.  
  
**7. Conclusion**Summarize the report.

(Most of function will be used in this web application are very identical to me and Guo’s previous project, the current idea realization and project development could be quiet feasible.)

**8. References**