

Team name: Team 28
Game name: Temporary

Team ideas:

- 28 studios
- Pufferfish Games
- Fugu
- 28th element - nickel
- Dime
- 28 in ascii - file separator control char
- 0x28 in ascii - (

:

Game name ideas:

- Heslington Hustle :(

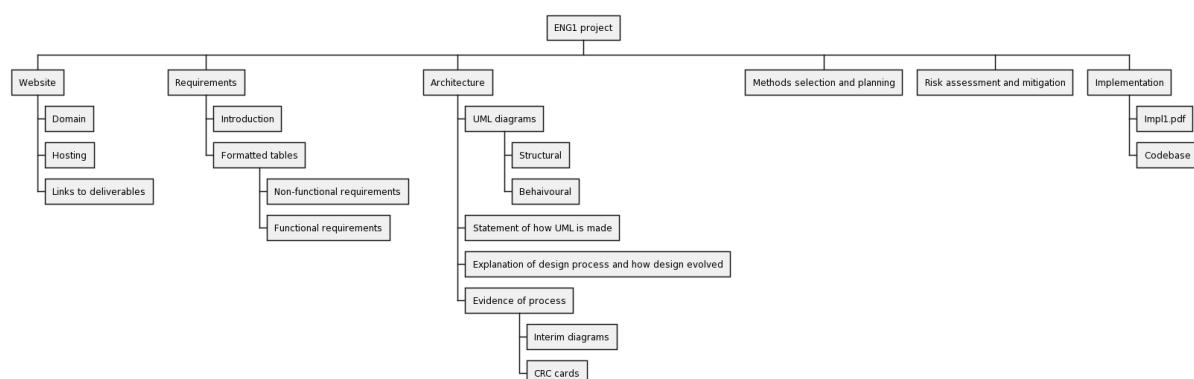
Meetings: whenever needed

Every monday:

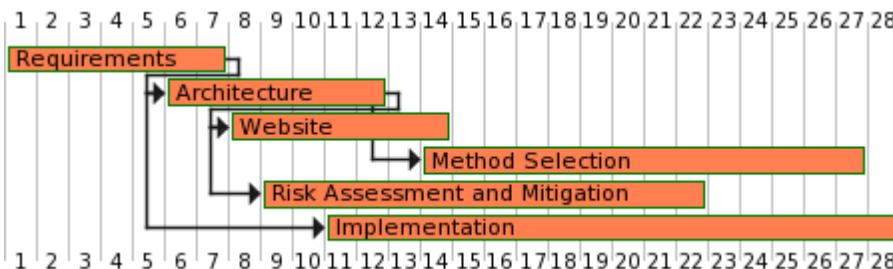
To do:

- Name team
- What platforms do we want to target - desktop definitely, web browser, android, and iOS all possible
- Get a free domain name for the website from github students. Need to be sure we're happy with the name e.g. team28.tech. Discuss at meeting
- Set up deliverable documents and zipping
 - Lectures recommended these:
 - <http://www.projectlibre.com/>
 - <https://plantuml.com/>
- Choose a license for game and website
- Choose website theme
- Formatter and linter - checkstyle, google code format, spotbugs, intellij?

plantuml task diagram



plantuml gantt chart



Game engine options

- Libgdx is the best one
- Fxgl also exists
- Slick2d seems dead?
- Java no libraries standard library
- JWJGL is the low level framework the other engines use
- Jmonkeyengine is 3d

Libgdx selected!

- Wiki: <https://libgdx.com/wiki/>
- Javadoc: <https://javadoc.io/doc/com.badlogicgames.gdx>

How to use project:

- Gradle command line
- IntelliJ (installed on uni computers) or android studio (version of IntelliJ with android build help, recommended by libgdk)
- Eclipse (recommended by course, bad)
- Vscode extensions

Java documentation: <https://docs.oracle.com/en/java/javase/11/docs/api/index.html>

Website

- GitHub pages docs: <https://docs.github.com/en/pages>
- Jekyll docs: <https://jekyllrb.com/docs/>

I'm experienced with both Git and GitHub Pages feel free to ask me about either on the discord - Joel

Github organisation: <https://github.com/eng1team28>

For sharing assets, the brief etc.

Google drive folder: [ENG1 Team 28](#)

- Req1
- Arch1
- Plan1
- Risk1
- Impl1

Deliverable Max mark Page limit File name and format

1. Website (submit only the URL) 3 — url1.txt
 2. Requirements 20 1 + 3 Req1.pdf
 3. Architecture 22 6 Arch1.pdf
 4. Method selection and planning 10 2 + 1 + 2 Plan1.pdf
 5. Risk assessment and mitigation 10 1 + 3 Risk1.pdf
 6. Implementation 25 1 Impl1.pdf
- + Code
+ Executable JAR

Inspiration - stick rpg

Fishing minigame

For minigames - bullet hell undertale style?

Team Logo - Temp



Website (3 marks) - End of W3/4

Requirements (20) - End of W2

Architecture (22) - End of W3

Method Selection and Planning (10) - End of W4

Risk Assessment and Mitigation (10) - End of W4

Implementation (25) - End of W4

Constraints

- Java ...
- Appropriately-licensed 3rd-party libraries, tools, and assets
- Accommodate stakeholder

Assessment 1 Requirements (from Product Brief)

1. Game Duration: The game will last for 7 days, with each day ending when the player goes to sleep. The game ends after the 7th day.
2. Game Avatar: The player will control an avatar representing a second-year computer science student. The avatar can move around the map and interact with various objects.
3. Control Scheme:
 - Moving the avatar around the map.
 - Interacting with what is in front of the avatar.
4. Activity Locations: Implement one of each interaction/activity location:
 - One place to sleep (e.g., a student accommodation building).
 - One place to study (e.g., the CS building or the university library).
 - One place to eat (e.g., the Piazza building or restaurants in town).
 - One recreational activity (e.g., going to the sports centre, spending time at the lake feeding ducks, etc.).
5. Map: The game must contain a map for the avatar to move around in. This map should contain at least some of the Heslington East campus (CS building, student accommodation building, Piazza building, etc.).
6. Activity Tracker: Implement a tracker for the 7 days the game lasts, with a mechanism to move to the next day.(sleeping?) The tracker should also include a simple counter denoting how many of each activity has been performed so far.
7. Game Objective: The primary objective of the game is to ensure the avatar has studied enough for exams, taken some time for themselves, and had enough rest every day.
8. Resource Consumption: Each activity will consume a varying amount of the avatar's two available resources: time and energy (X%). Each day contains 16 hours, and each day the avatar has an energy bar/capacity.