

The background features a complex network of thin grey lines and dots, forming a web-like structure. Scattered throughout are various triangles of different sizes and orientations, some with solid black dots at their vertices. The overall aesthetic is technical and minimalist.

TECHNICAL DESIGN DOCUMENT

Enric-G. Durán Vilar

01

THE PROBLEM



THE PROBLEM

The TDD answers to a specific design need in a project,

A NEED...

- To design the execution of the GDD's proposals
- To organise the code and project structure
- To establish software and hardware tools, platforms and limitations
- To have a common tool that establishes the scope of the project



02

APPROACHES



APPROACHES



SOFTWARE

In essence, the template for any TDD follows the same logical order that goes from what we want to do to what platform do we do it for, how we do it, where we do it and how we deliver it.

Our project's scope is very different from big productions and there are technical differences between genres, platforms, artstyles...

GAMES



THIS SUBJECT

A situation nearer to the scope of our own project. The point is NOT to copy, but to learn and improve.

***Everything has been done before!**



03

MY SOLUTION

INTRODUCTION

Name of the game, copyright and a couple lines about the project

01

TECHNICAL GOALS

Technical specifications and limitations of the equipment your game's designed to thrive in

02

DEV HARDWARE

Technical specifications and limitations of the equipment your team will thrive in

03

DEV SOFTWARE

Code and environments your team will work with for the development

04

GAME MECHANICS

Technical implications of every part of the game's design

05

TDD STRUCTURE

06

CODE STYLE

Standards and conventions for the style of your team's code

07

UML

Code Organization Overview UML that explains the whole project

08

DATA LAYOUT

Folder organisation of your project

09

SCHEDULING

Requirements for each delivery made by the team in the future

10

DELIVERY METHOD

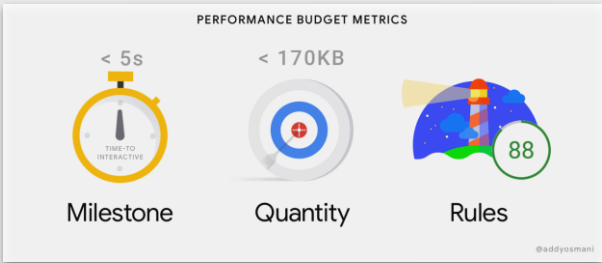
How will you make builds and deploy them?

01 INTRODUCTION

- Name of the game
- Team Members
- Brief description of information of interest (genre, feel, mechanism)
- Code Language
- Licensing

02 TECHNICAL GOALS

- Target Platform Specs
- Performance budgets



Main processor	Single-chip custom processor
	CPU : x86-64 AMD "Jaguar", 8 cores
	GPU : 1.84 TFLOPS, AMD Radeon™ based graphics engine
Memory	GDDR5 8GB
Storage size*	500GB, 1TB
BD/ DVD drive (read only)	BD × 6 CAV DVD × 8 CAV
Input/ Output	Super-Speed USB (USB 3.1 Gen1) port × 2 AUX port × 1
Networking	Ethernet (10BASE-T, 100BASE-TX, 1000BASE-T) ×1
	IEEE 802.11 a/b/g/n/ac
	Bluetooth®v4.0
Power	AC 100-240V, 50/60Hz
Power consumption	Max. 165W

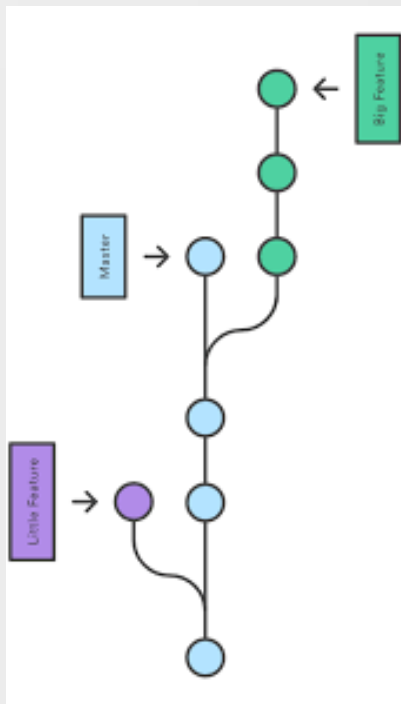
03 DEVELOPMENT HARDWARE

WHAT ARE WE MAKING THE GAME WITH?

- Computers
 - ◀ Technical specs
 - ◀ OS
 - ◀ Program runtime
- Screens
 - ◀ Resolution
- Art Tools
 - ◀ Graphic Tablets
 - ◀ Cameras
 - ◀ Computers
 - ◀ ...



04 DEVELOPMENT SOFTWARE



ART



Pre-existing tools that we will
rely on to make our project

TECHNICAL TOOLS/LIBRARIES



ENGINES



CRYENGINE

05 GAME MECHANICS



TECH REQUIREMENTS

List of every feature that is wished to be implemented by the end of the development



GAME ARCHITECTURE

The definition of the flow of the game and changes between its different screens, as well as the data management



GRAPHICS & AUDIO

Graphics & Audio specifications and limits, implementation in the game



AI

Explanation of the behaviour of the entities that require some sort of AI programming



LOGIC

Explains the different Game Objects and how they are structured



PHYSICS & COLL.

Where collisions are collisions, what sort of physics the game implements

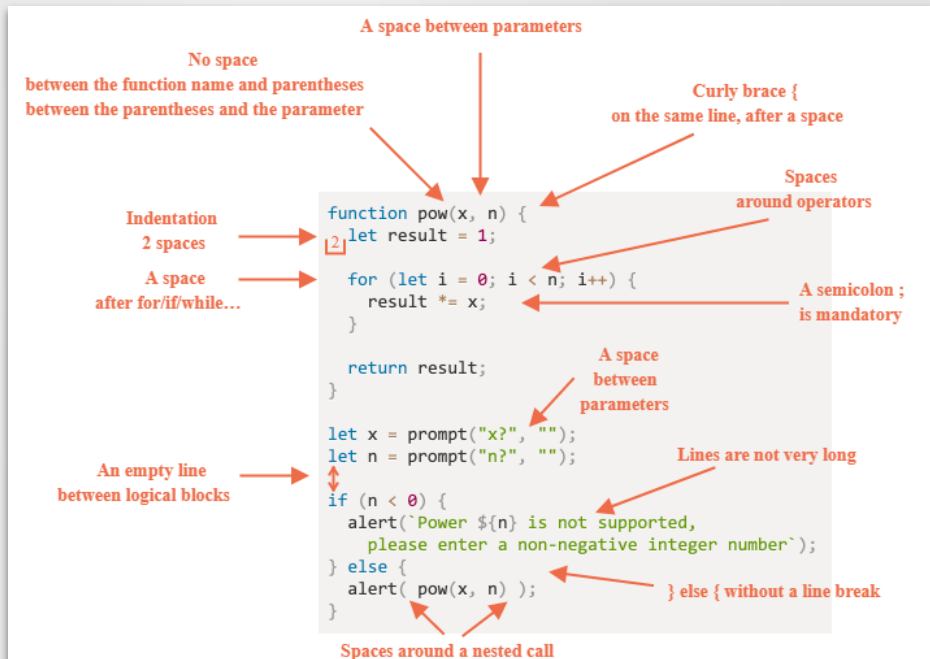
***Network**



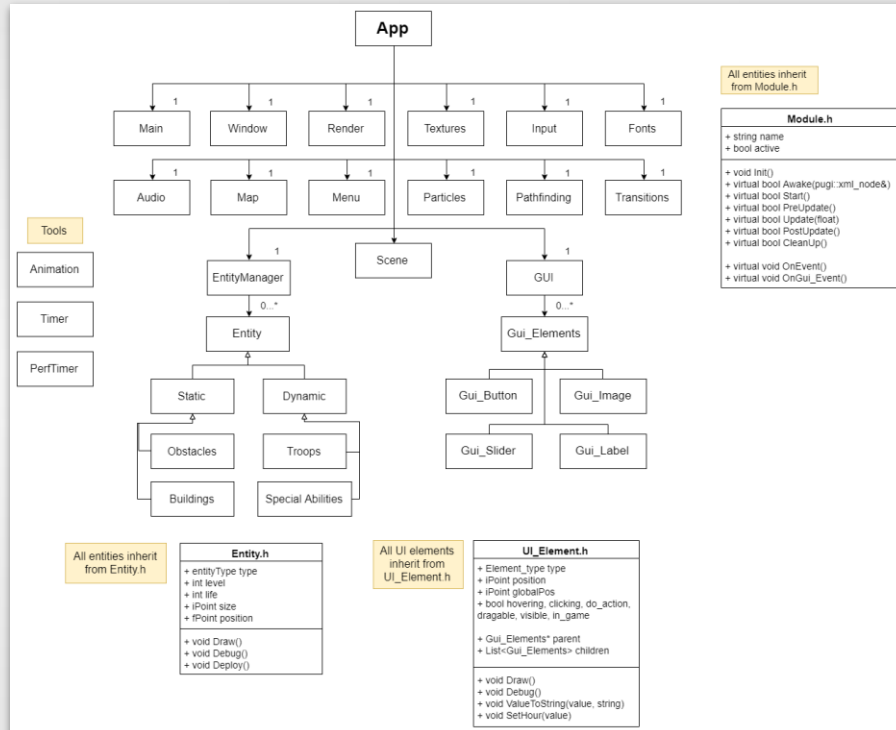
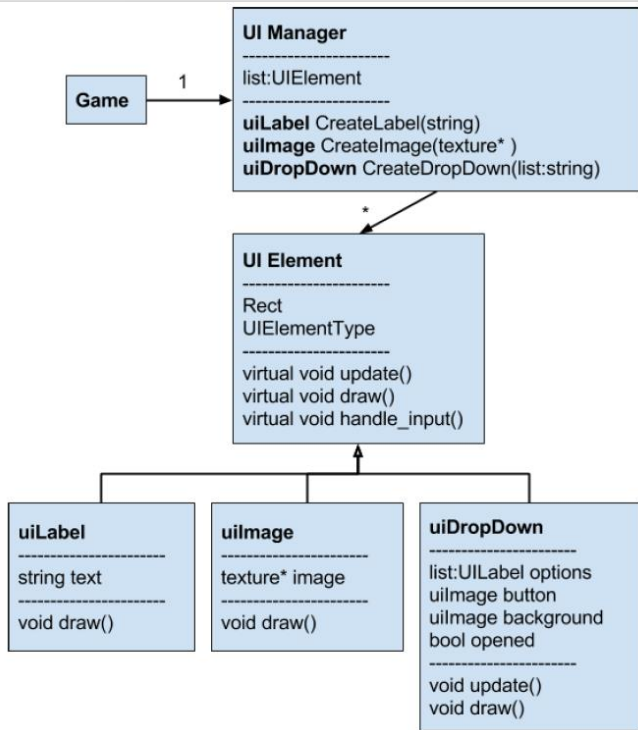
06 CODE STYLE

HOMOGENEOUS CODE IS MORE READABLE CODE!

- Naming
- Variables
- Loops
- Conditionals
- Classes & Structs
- XML



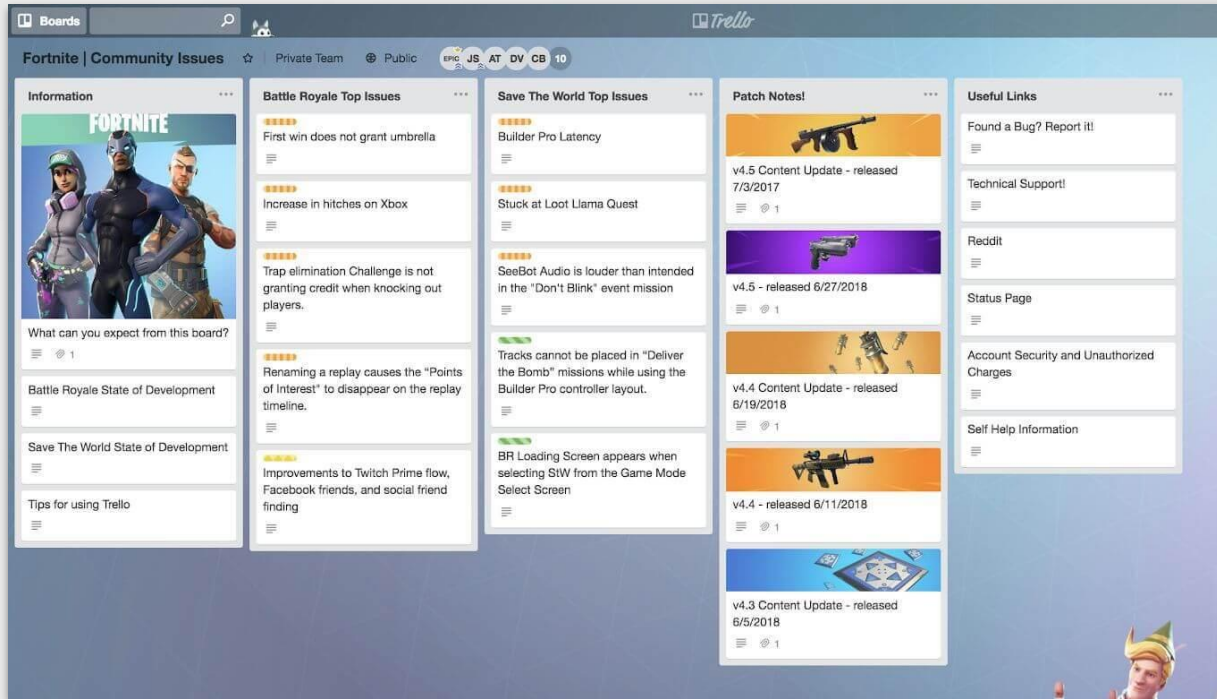
07 UML



08 DATA LAYOUT

- ▼ PaNOSC
 - ▼ 1 Work Packages
 - ▼ WP1 Management
 - 1 Stakeholders
 - 2 Plans
 - 3 Risks
 - 4 Issues
 - ▼ 5 Meetings
 - ▼ 1 Executive Committee
 - YYYY-MM-dd Executive Committee ##
 - > 2 Project Management Committee
 - 3 Other Meetings
 - WP2 Data Policy and stewardship
 - WP3 Data catalogue services
 - WP4 Data analysis services
 - WP5 Virtual Neutron and X-ray Laboratory
 - WP6 EOSC Integration
 - WP7 Sustainability
 - WP8 User Training
 - ▼ WP9 Outreach and communication
 - 1 Presentations
 - ▼ 2 Submitted Deliverables
 - ▼ 1 Pre Project
 - 01 Proposal
 - 02 Grant Agreement
 - 03 Consortium Agreement
 - ▼ 2 Project
 - D1.1
 - D1.2
 - D9.3
 - 3 Post Project
 - ▼ 3 Publications
 - YYYY-MM-dd [NAME]

09 SCHEDULE



10 BUILD DELIVERY METHOD

The screenshot displays the GitHub Actions interface for a workflow named "hello-world-workflow". The interface is divided into a left sidebar and a main content area. The sidebar shows the workflow status as "Update main.yml" with a green checkmark. Below this, the workflow is expanded to show a list of jobs: "Hello world workflow on: push", "A job run on Linux" (selected), and "A job run in a container". The main content area shows the details of the selected job, "A job run on Linux", which is successful. The job steps are listed with their durations: "Set up job" (2s), "Run actions/checkout@master" (2s), "Hello world action step" (1s), "Get the output time" (0s), and "Complete job" (0s). The "Hello world action step" is expanded to show its log output, which includes a timestamp and the text "The time was 01:38:44 GMT+0000 (Coordinated Universal Time)".

Workflow status

Last run

Workflow / Check suite

Job / Check run

Step

Download logs / More options

Run time

Log view

04

EXERCISE



INTRODUCTION

Name of the game, copyright and a couple lines about the project

01

TECHNICAL GOALS

Technical specifications and limitations of the equipment your game's designed to thrive in

02

DEV HARDWARE

Technical specifications and limitations of the equipment your team will thrive in

03

DEV SOFTWARE

Code and environments your team will work with for the development

04

GAME MECHANICS

Technical implications of every part of the game's design

05

TDD STRUCTURE

06

CODE STYLE

Standards and conventions for the style of your team's code

07

UML

Code Organization Overview UML that explains the whole project

08

DATA LAYOUT

Folder organisation of your project

09

SCHEDULING

Requirements for each delivery made by the team in the future

10

DELIVERY METHOD

How will you make builds and deploy them?



05

BIBLIOGRAPHY

Bibliography

- Last year's research
- TDD Purpose
- Software TDD 1
- Game TDD
- TDD Example 1
- TDD Example 3
- Branching in GitHub





THANK YOU FOR YOUR ATTENTION

Any questions?

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**.

Please keep this slide for attribution.