



ΑΝΑΠΤΥΞΗ ΨΗΦΙΑΚΩΝ ΠΑΙΧΝΙΔΙΩΝ ΚΑΙ ΠΟΛΥΜΕΣΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΤΜΗΜΑ ΕΠΙΚΟΙΝΩΝΙΑΣ ΚΑΙ ΨΗΦΙΑΚΩΝ ΜΕΣΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ

Επιβλέπων καθηγητής:

Δρ. Μηνάς Δασυγένης

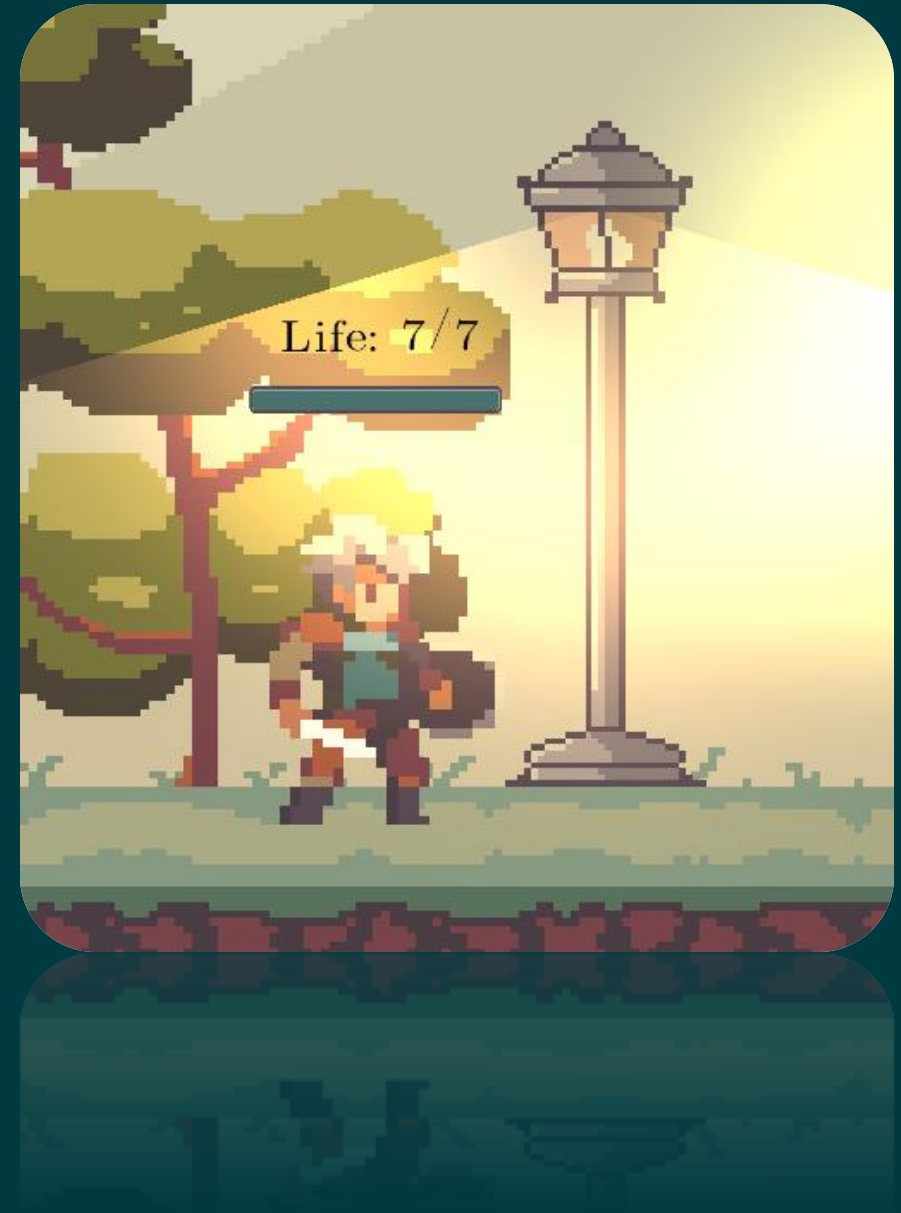
ΜΗΧΑΝΗ ΔΗΜΙΟΥΡΓΙΑΣ ΠΑΙΧΝΙΔΙΩΝ (ΧΩΡΙΣ ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ): ΑΡΧΙΤΕΚΤΟΝΙΚΟΣ ΣΧΕΔΙΑΣΜΟΣ, ΥΛΟΠΟΙΗΣΗ, ΔΗΜΙΟΥΡΓΙΑ ΠΑΡΑΔΕΙΓΜΑΤΩΝ ΠΑΙΧΝΙΔΙΩΝ

Κωνσταντίνος Γούλας

Ιούνιος 2026

ΠΕΡΙΕΧΟΜΕΝΑ

- Κίνητρο της εργασίας
- Προδιαγραφές και στόχοι
- Τεχνική υλοποίηση της μηχανής παιχνιδιών
- Δημιουργία παιχνιδιών
- Συμπεράσματα και μελλοντικές επεκτάσεις



ΚΙΝΗΤΡΟ ΤΗΣ ΕΡΓΑΣΙΑΣ

Κατανόηση της αρχιτεκτονικής μιας σύγχρονης μηχανής παιχνιδιών.

Οι σύγχρονες εμπορικές μηχανές:

- Πολλές δυνατότητες.
- Αποκρύπτουν την πολυπλοκότητα.

Η μηχανή που αναπτύχθηκε για την εργασία:

- Ελαφριά υλοποίηση.
- Διακριτά συστήματα και υποσυστήματα.



ΣΤΟΧΟΙ ΚΑΙ ΠΡΟΔΙΑΓΡΑΦΕΣ

Στόχοι:

- Μελέτη της αρχιτεκτονικής μιας μηχανής παιχνιδιών
- Σχεδίαση μιας "code-first" μηχανής παιχνιδιών από το μηδέν
- Αξιολόγηση μέσω ανάπτυξης παιχνιδιών
- Διερεύνηση χρήσης τεχνητής νοημοσύνης (AI) στην ανάπτυξη παιχνιδιών

Προδιαγραφές:

1. Φορητή (Cross Platform) και επεκτάσιμη εφαρμογή:
 - Γλώσσα C++ (με OpenGL και GLFW)
 - Python (API), pybind11 library
2. Χωρίς γραφικό περιβάλλον
3. Για δημιουργία 2D παιχνιδιών
4. Με όλα τα βασικά συστήματα που απαιτεί μια σύγχρονη μηχανή παιχνιδιών
5. Χρήση Τεχνητής Νοημοσύνης για την ευκολότερη δημιουργία παιχνιδιών

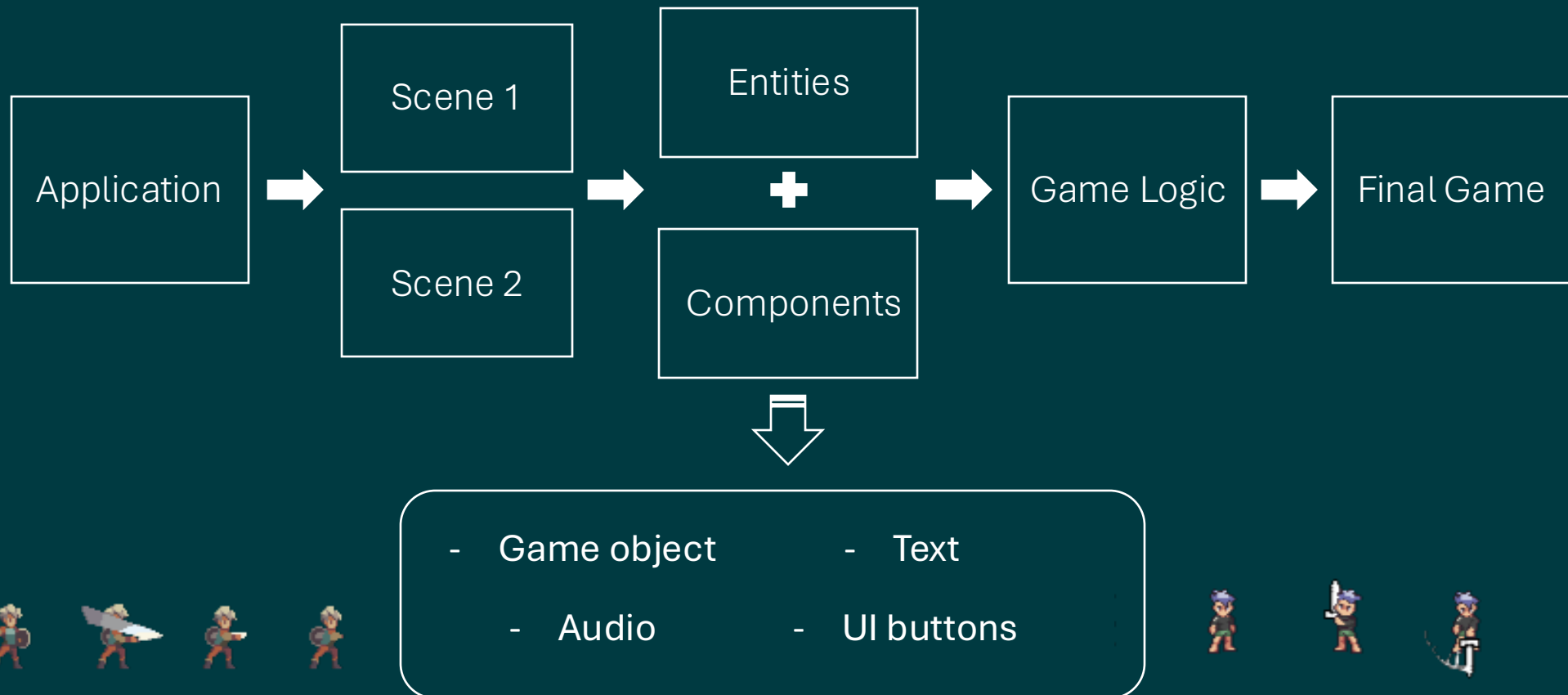
ΤΑ ΥΠΟΣΥΣΤΗΜΑΤΑ ΠΟΥ ΚΑΛΥΠΤΕΙ Η ΜΗΧΑΝΗ



Game Engine Typical Systems	My Engine Systems
ECS	✓
Scene System	✓
UI Manager	✓
3D Engine + 3D Physics	✗
2D Engine + 2D Physics	✓
Audio Manager	✓
Network System	✗



Η ΡΟΗ ΔΗΜΙΟΥΡΓΙΑΣ ΕΝΟΣ ΠΑΙΧΝΙΔΙΟΥ



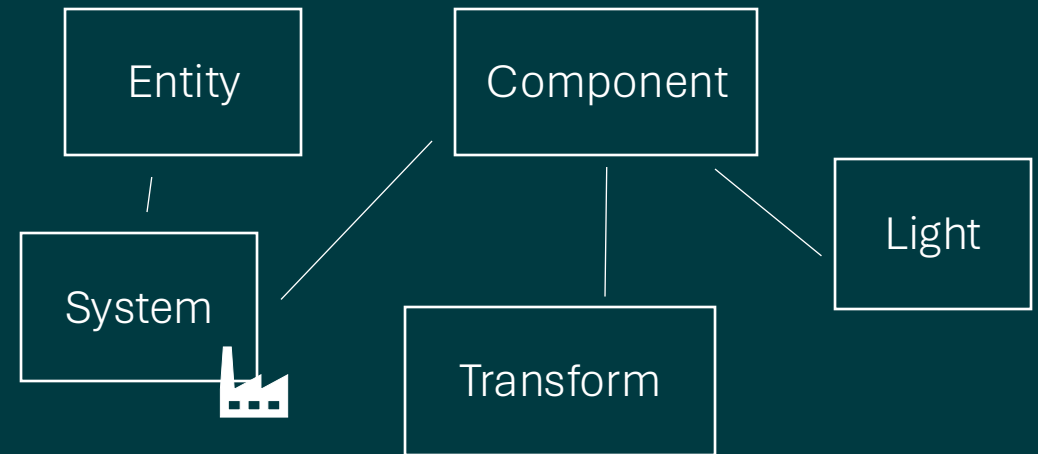
ΓΙΑΤΙ ΕΠΙΛΕΧΘΗΚΕ ΤΟ ENTITY COMPONENT SYSTEM (ECS)

Αρχική προσέγγιση:

1. Κλάση Αντικειμένου παιχνιδιού (Game object), που "κουβαλάει" τη γεωμετρία (3D mesh)
2. Κλάση Sprite (2D, texture, animation)

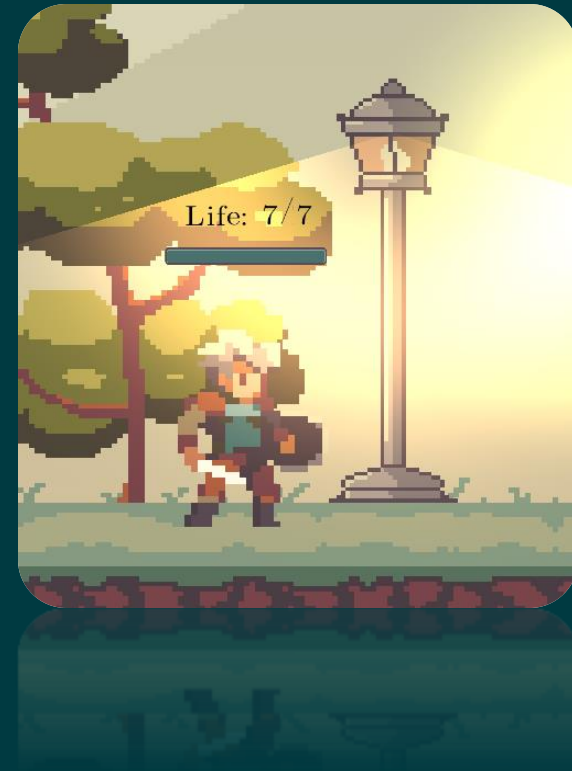
Τελική προσέγγιση (ευελιξία):

1. Μοναδική ταυτότητα (Entity)
2. Κλάση δεδομένων (Component)
3. Κλάση επεξεργασίας των δεδομένων (System)



ΥΛΟΠΟΙΗΜΕΝΑ COMPONENTS ΤΟΥ ECS

1. Transform Component
2. Sprite Component (οπτική αναπαράσταση 2D οντοτήτων)
3. Light Component
4. Box Collider Component
5. Camera Component
6. Audio Component
7. Text Component

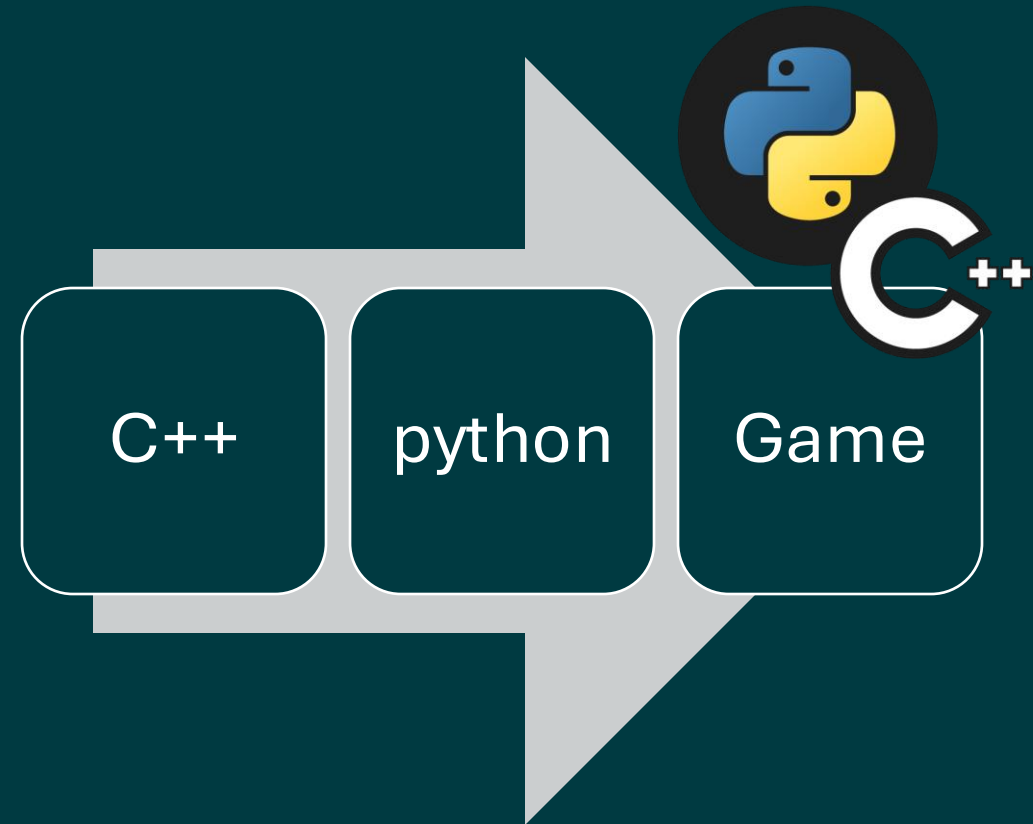


ΓΙΑΤΙ ΠΡΟΣΤΕΘΗΚΕ PYTHON (API)

1. Βιβλιοθήκη pybind11 (C++ 11 και νεότερες εκδόσεις)

Από C++ στην Python :

1. Ευκολότερη υλοποίηση του παιχνιδιού
2. Απόκρυψη συστημάτων (που δεν χρειάζεται άμεσα ο χρήστης)



ΕΠΙΠΛΕΟΝ ΥΠΟΣΥΣΤΗΜΑΤΑ ΤΗΣ ΜΗΧΑΝΗΣ

- Κλάση της κύριας εφαρμογής (Application class).
- Σύστημα δημιουργίας σκηνής (Scene).
- Σύστημα διαχείρισης διεπαφής χρήστη (UI Manager).
- Σύστημα διαχείρισης εισόδου-εξόδου (Input – Output System)
- Μαθηματικά εργαλεία
 - AABB2d axis-aligned bounding box
 - intersection algorithms
 - mesh structure



ΠΑΡΑΔΕΙΓΜΑ 1: 2D ΠΑΙΧΝΙΔΙ ΔΡΑΣΗΣ

- Python αρχεία: 9
- Γραμμές κώδικα: 1166

Run – Jump – Hit – Idle



ΤΙ ΔΟΚΙΜΑΣΤΗΚΕ ΣΤΟ ΠΡΩΤΟ ΠΑΙΧΝΙΔΙ

1. Application class
2. Event System
3. Scene class
4. ECS (Transform, Light, Text, Audio, Box Collider, Sprite, Camera)
5. UI Manager

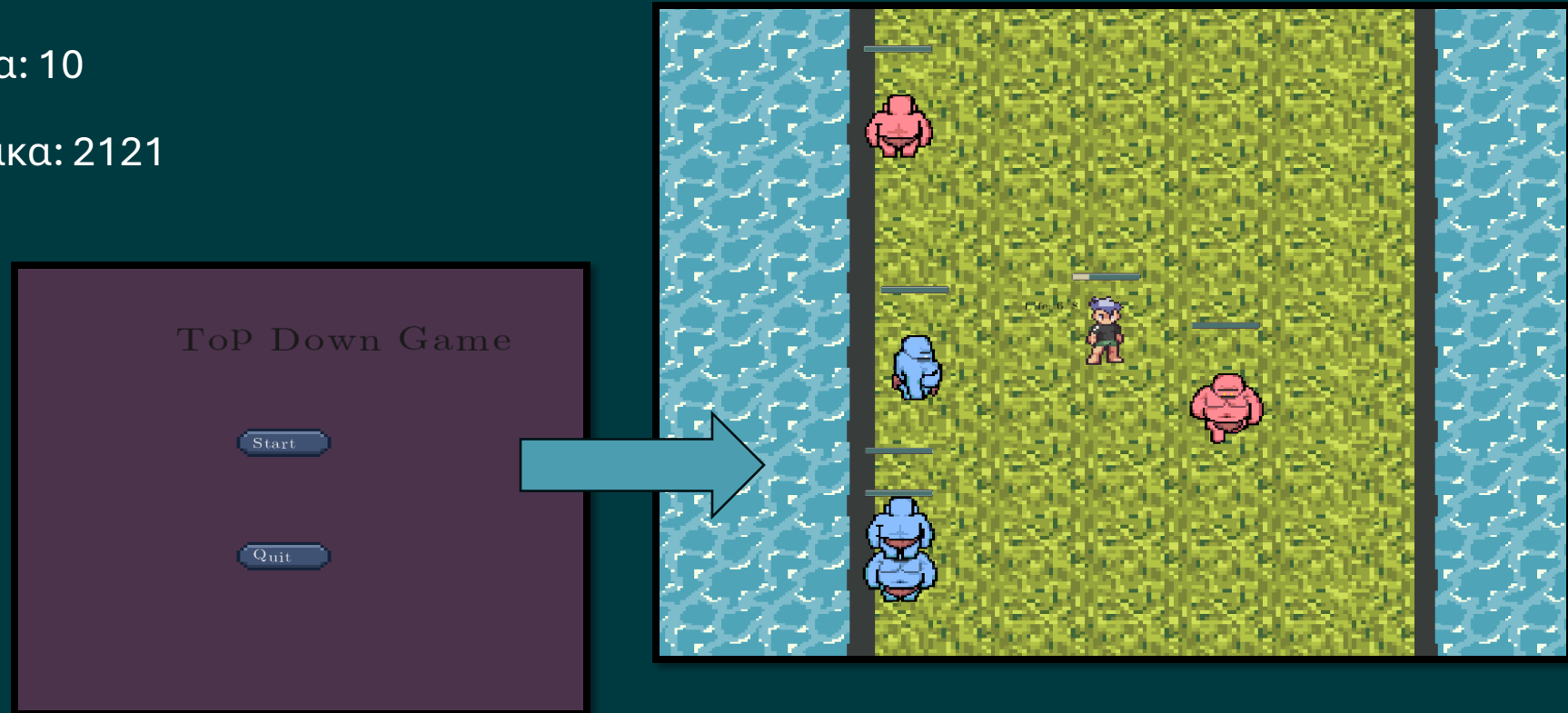


DEMO VIDEO 1: 2D ΠΑΙΧΝΙΔΙ ΔΡΑΣΗΣ



ΠΑΡΑΔΕΙΓΜΑ 2: TOP-DOWN 2D ΠΑΙΧΝΙΔΙ

- Python αρχεία: 10
- Γραμμές κώδικα: 2121



ΤΙ ΠΡΟΣΤΕΘΗΚΕ ΣΤΟ TOP-DOWN ΠΑΙΧΝΙΔΙ

1. Application class
2. Event System
3. Scene class
4. ECS (Transform, Light, Text, Audio, Box Collider, Sprite, Camera)
5. UI Manager
6. Tile map class



DEMO VIDEO 2: TOP-DOWN ΠΑΙΧΝΙΔΙ



ΠΕΙΡΑΜΑ: ΑΝΑΠΤΥΞΗ ΠΑΙΧΝΙΔΙΟΥ ΜΕ ΑΙ ΜΟΝΤΕΛΑ

Μοντέλα:

- Claude 3.5. Sonnet
- Claude Haiku 4.5.
- Big Pickle (by OpenCode)

Προδιαγραφές:

- Κοινό αρχικό prompt
- Καμία άμεση παρέμβαση στον κώδικα
- Καθαρό περιβάλλον πειράματος



ΠΑΙΧΝΙΔΙ ΠΕΙΡΑΜΑΤΟΣ: SPACE SHOOTER



Περιγραφή του παιχνιδιού:

- Space shooter
- Κίνηση δεξιά - αριστερά
- Εξολόθρευση αντιπάλων
- Περιορισμένες σφαίρες



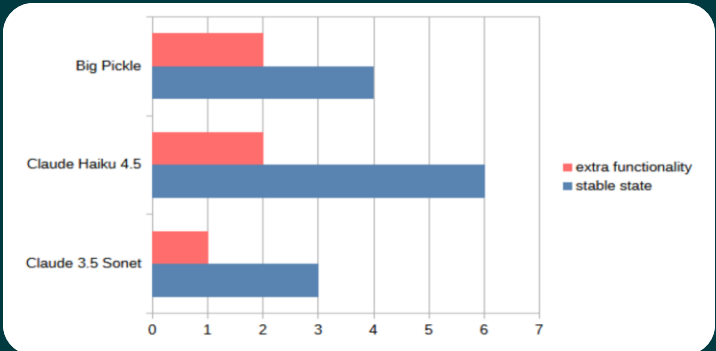
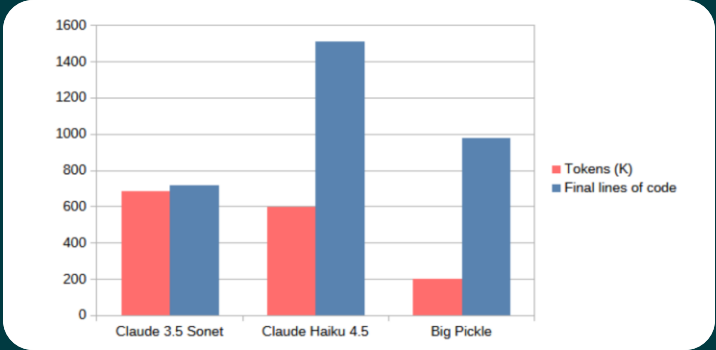
ΣΥΓΚΡΙΣΗ ΠΕΙΡΑΜΑΤΩΝ ΤΩΝ ΑΙ ΜΟΝΤΕΛΩΝ

1

3

2

Μοντέλο	Claude 3.5 Sonnet	Claude Haiku 4.5	Big Pickle
Οπτικά προβλήματα	✓	✓	✓
Αλγοριθμικά προβλήματα	✗	✓	✗
Κόστος (per million input tokens)	\$2	\$1	\$0
Κόστος (per million output tokens)	\$10	\$5	\$0
Συνολικό κόστος	\$2	\$0.9	\$0



DEMO VIDEO 3: SPACE SHOOTER



ΣΥΜΠΕΡΑΣΜΑΤΑ

Επιτεύχθηκαν οι στόχοι :

- ✓ Δημιουργήθηκε μια ελαφριά "code-first" μηχανή παιχνιδιών.
- ✓ Δημιουργήθηκαν παιχνίδια (διαφορετικού τύπου) με χρήση της μηχανής.
- ✓ Τα AI μοντέλα κατάφεραν να προσαρμοστούν και να υλοποιήσουν εφαρμογές με βάση τη μηχανή.



ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

- Τρισδιάστατα γραφικά
- Δικτυακή λειτουργία
- Ανάπτυξη γραφικού περιβάλλοντος χρήστη
- Περαιτέρω διερεύνηση της χρήσης τεχνητής νοημοσύνης

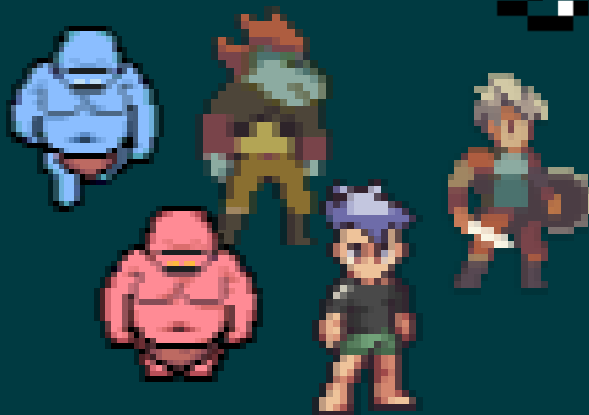




ΑΝΑΠΤΥΞΗ ΨΗΦΙΑΚΩΝ ΠΑΙΧΝΙΔΙΩΝ ΚΑΙ ΠΟΛΥΜΕΣΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΤΜΗΜΑ ΕΠΙΚΟΙΝΩΝΙΑΣ ΚΑΙ ΨΗΦΙΑΚΩΝ ΜΕΣΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ

THANK YOU



Κωνσταντίνος Γούλας

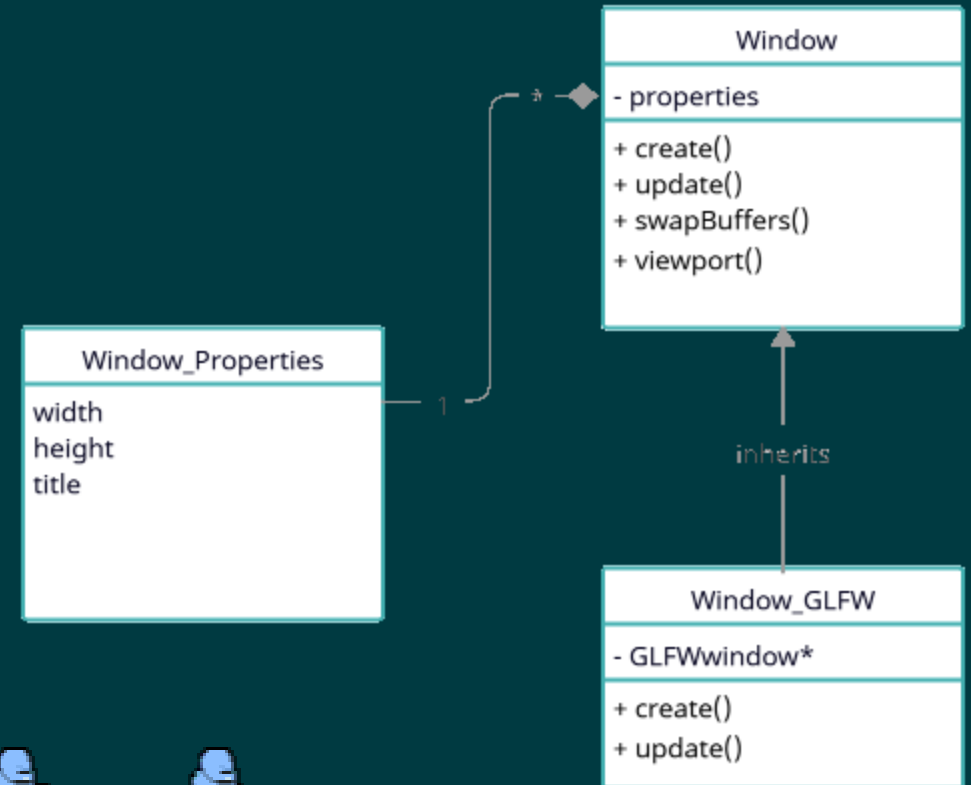
kzgoulas@gmail.com

ΠΑΡΑΡΤΗΜΑ

ΔΗΜΙΟΥΡΓΙΑ ΠΑΡΑΘΥΡΟΥ

Από "κάτω" προς τα "πάνω" :

- GLFW (επικοινωνία με το hardware).
- Window class (C++) στον πυρήνα της μηχανής.
 - o Window properties: τα δεδομένα του παραθύρου.
- Window class (Python): η κλάση που μπορεί να χρησιμοποιήσει ο χρήστης.



ΔΙΑΧΕΙΡΙΣΗ ΕΙΣΟΔΟΥ

Στη βάση του συστήματος βρίσκεται η GLFW.

Πάνω σε αυτήν χτίζουμε:

1. Μια κωδικοποίηση των δεδομένων εισόδου (enum class).
2. Σύστημα διαχείρισης γεγονότων (event system).
 - ο Κλάση γεγονότων (Event class).
 - ο Κλάση διαχείρισης γεγονότων (Event Dispatcher).



ΑΠΟΔΟΣΗ ΓΡΑΦΙΚΩΝ

1. Ορισμός γεωμετρίας (Vertices).
2. Μεταφορά δεδομένων στην κάρτα γραφικών (GPU), μέσω "buffer objects".
3. Πρόγραμμα επεξεργασίας "κορυφών" (Vertex Shader).
4. Ραστεροποίηση (Rasterization).
5. Πρόγραμμα επεξεργασίας "τμημάτων" (Fragment Shader).
6. Γέμισμα του τελικού πίνακα.

