

Relazione Sprint 0

Team 9

October 26, 2024

1 Generali

Come team abbiamo scelto di sviluppare il Bird-Sort-clone, una app di puzzle per mobile.

Utilizzeremo l'engine open source **Godot**, programmando in **GDScript** per il front-end e in **Python** per il back-end. Abbiamo stabilito l'utilizzo di *snake_case* per i nomi di variabili e di *CamelCase* per i nomi delle classi, evitando abbreviazioni che riducono la leggibilità. Ci impegneremo a scrivere *docstrings* per tutte le funzioni dove siano necessarie, a scopo di chiarire lo scopo e l'utilizzo della funzione. L'utilizzo di commenti inline sarà moderato ma presente. Puntiamo all'utilizzo di eccezioni per l'*error handling*, scegliendo tipi di eccezione specifici invece che generali. GDScript è event-driven, quindi sarà importante mantenere chiarezza nella nomenclatura dei *signal*.

Il lavoro svolto nello Sprint 0 è stato svolto principalmente da remoto, con un meetup per giocare a Scrumble di persona. Il canale di comunicazione usato fino a questo punto è stato una chat di gruppo su Whatsapp, scelta per familiarità e convenienza, ma possibilmente cambierà in futuro.

2 Ruoli del gruppo

- **Thomas Stephens:** Product Owner
 - Sottoruolo: UX
- **Gianluca Noviello:** Scrum Master
 - Sottoruolo: UX
- **Angelo Ruggieri:** Developer
 - Sottoruolo: Front
- **Ricardo Muzzi:** Developer
 - Sottoruolo: Design
- **Gian Luca Montefiori:** Developer
 - Sottoruolo: Front
- **Piero Mongelli:** Developer
 - Sottoruolo: Back
- **Erik Dervishi:** Developer
 - Sottoruolo: Back

3 Tempistica

La tempistica del lavoro, per quanto riguarda questo Sprint 0, si riferisce allo Scrum Master e il Product Owner. Al di fuori di Scrumble, dove hanno contribuito tutti, il lavoro è principalmente stato svolto da i due ruoli.

3.1 Setup CAS

Il setup dell'environment digitale che andremo a usare si è manifestato di continuo durante le 2 settimane dello Sprint 0, in quanto ogni membro stava aspettando l'abilitazione dell'account. La creazione del Backlog su Taiga ha impiegato un pomeriggio, ma ha subito varie rivisite con l'andare del tempo.

3.2 Scrumble

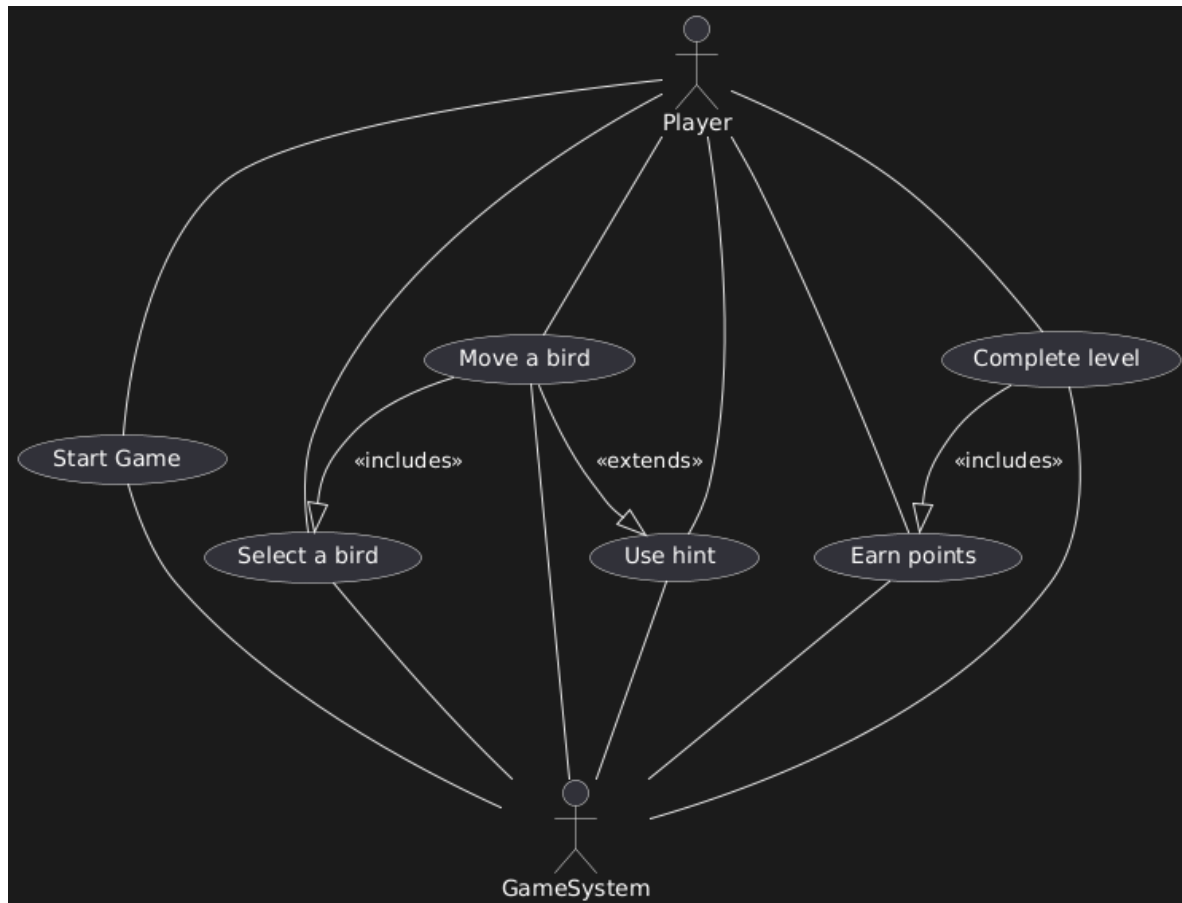
Dopo una settimana di pianificazione ci siamo ritrovati di persona per giocare a Scrumble, avendo deciso di stampare la tavola da gioco, le pedine e tutto il necessario. Dopo 30 minuti impiegati per capire le regole collettivamente, abbiamo iniziato a giocare verso le 10:30 del mattino. Contando una pausa pranzo di 1 ora, abbiamo terminato il gioco verso le 16:30. La relazione sulla partita ha impiegato all'incirca 2 ore, con altro tempo extra per revisioni aggiuntive. L'autovalutazione ha impiegato poco tempo, seguendo il template provvisto nel materiale condiviso.

3.3 UML

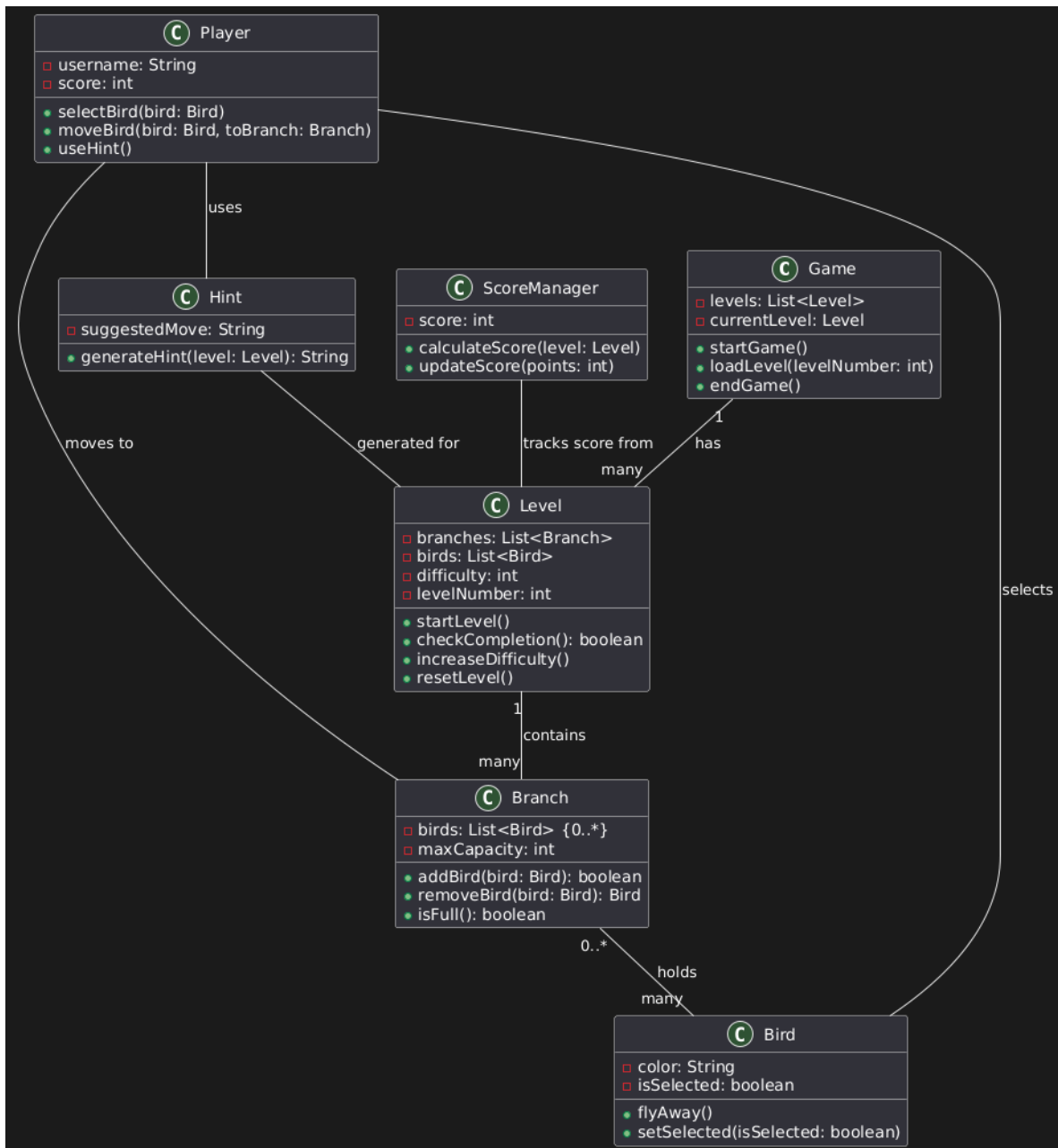
La creazione degli schemi UML ha impiegato 1 ora e 30 minuti circa.

4 Schemi UML

4.1 Schema Usecases



4.2 Schema Classi



4.3 Schema Statechart

