

02 Game Documentation

Tvorba a dizajn počítačových hier (FMFI)

Návrh a vývoj počítačových hier (FIIT)

Michal Ferko

28. 9. 2023

Documents related to Game Design

- Concept Document
- Game Proposal Document
- Game Design Document (GDD)
- Technical Design Doc
- Art Style Guide / Art Design Doc
- Project Plan & Schedule
- Test Plan

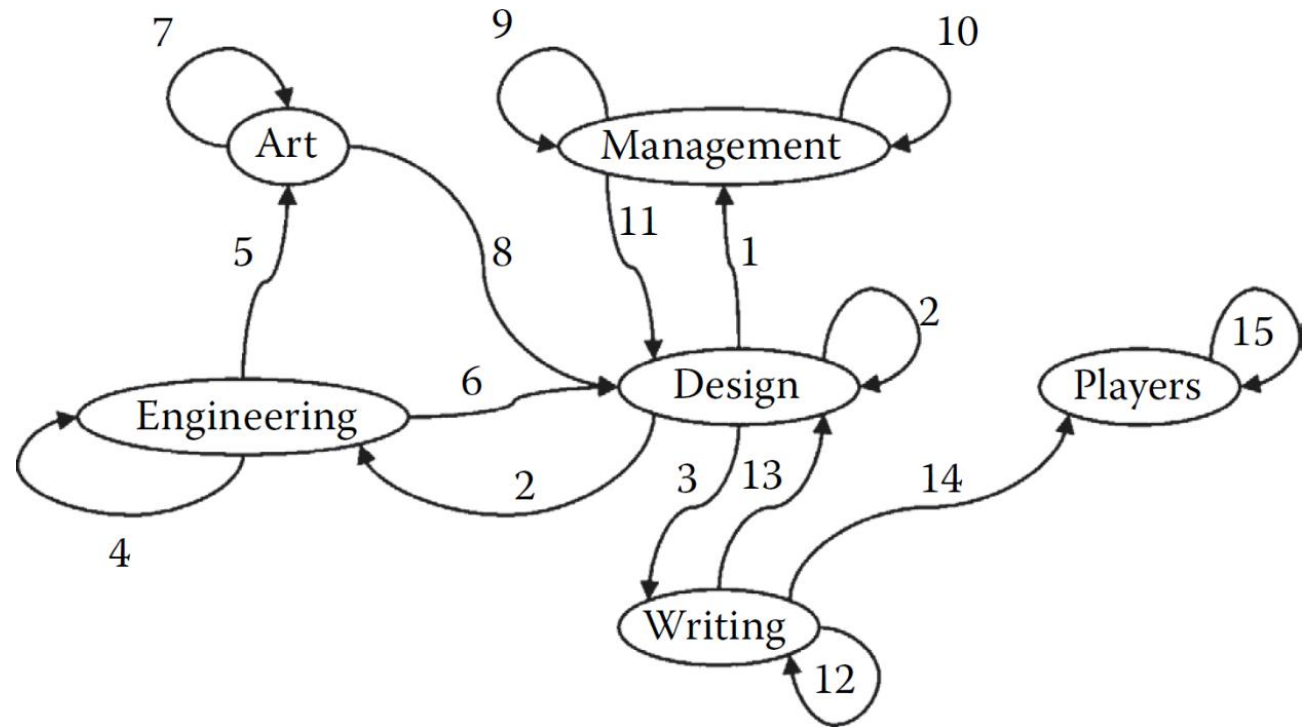


Image from *The Art of Game Design: A book of lenses*

Pitch Deck (or Concept Doc)

- Convey the goal and purpose of the game
- For management level to help “sell” the game
 - Assess viability, budget, scope & timeline
 - Sell to management/investors/publishers/co-founders
- Usually a presentation, with ~10 slides
 - Concept doc can be 1-2 pages, max. up to 5
- Written by the producer/creative director
- No well-defined form
 - List key things you want to communicate
 - Sort by importance
 - Most important first (since nobody might read the whole thing)

Pitch Deck contents

1. **Premise** – high level concept, describe your game and how it is unique in a few sentences
2. **Player motivation** – what will motivate the players? How do they win?
3. **Hook/Unique Selling Points** – Why will players pick YOUR game?
 - What makes it different from the rest
4. **Target Market** – Age, Gender, Motivators, Platform, Genre, HW requirements...
 - Target rating (mature content, violence, ...)
5. **Competitor & market analysis**
6. **Budget, scope, timeline, revenue projections**
7. **License**

Pitch Decks

- Loot River: Prototypovanie a príprava pitch decku | Miro Straka, Straka Studio
 - <https://www.youtube.com/watch?v=oJ8BJl3uxgQ>
- The 10-20-30 rule of PowerPoint
 - https://guykawasaki.com/the_102030_rule/
- Pitch Deck examples
 - <https://www.notion.so/Pitch-Decks-f56e38c13fe6417f8379859e74367e1a>
- How to pitch your game to a publisher or anyone
 - <https://www.tinybuild.com/how-to-pitch-your-game>

Game Design Document (GDD)

- Similar to a *software requirements specification*
- Complete description of the software
 - All functional requirements
 - Detailed description of features
 - Should answer all questions about the game
- The GDD must contain
 - Complete gameplay description – mechanics, interactions
 - Description of all game elements (creatures, NPCs, items, classes, story...)
- Can be extremely long – 50 – 200 pages
- It's a **living document**
 - Updated throughout development
 - When a design change occurs, the designer must update the GDD

GDD

- The GDD is used mainly by the production team
 - **Artists** when creating art
 - **Programmers** when programming features
 - **Designers** when creating new mechanics or improving existing ones
- Should be the “ground truth” for the team
 - When something is unclear, the GDD should have the answer
 - If it does not, the designer should update it
- The form is relatively loose
 - Certain sections are irrelevant for certain games
 - There is no magic template
- Purposes: **memory** and **communication**
 - You make lots of decisions and you might forget them

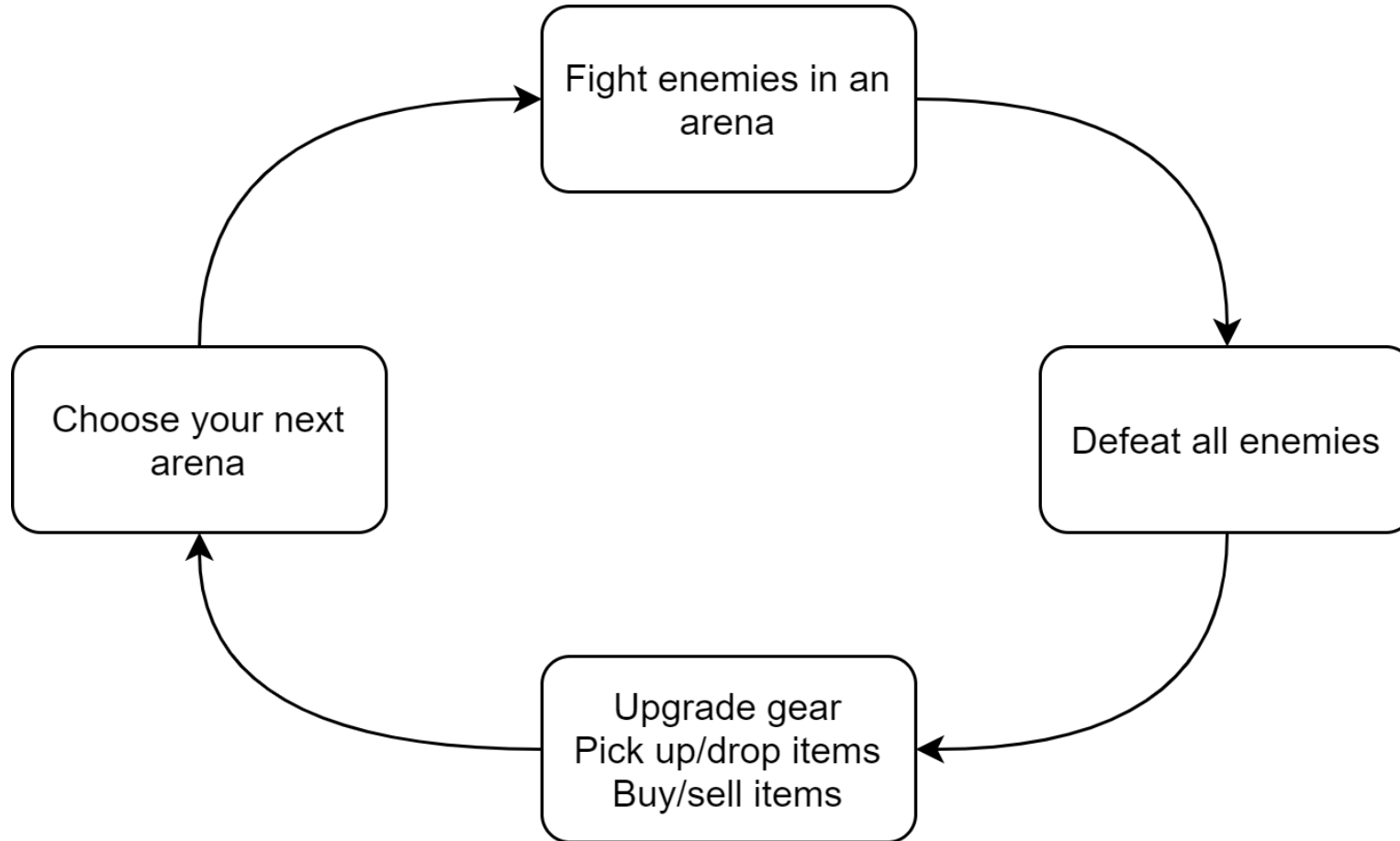
GDD forms

- Google Docs
 - **Pros:** Easily shareable, can link to any section, quick to prepare, all in one place
 - **Cons:** Struggles with larger documents (50+ pages with images), more linear reading
 - My experience: people use a ton of separate Google Docs and link between them
- Wiki system
 - **Pros:** More structured, can link very well, quickly editable, good revision history
 - **Cons:** Does not have everything in one place, slower setup, might be harder to edit/maintain
- Miro (single visual board)
 - **Pros:** Everything in one place, more visual, less structured, *able to integrate detailed documentation into diagrams*
 - **Cons:** If you do not maintain structure, it can be very confusing and hard to navigate

GDD Structure

- Core gameplay description
- Core game loop
- Character, Controls, Camera
 - Player tools
 - Challenges (problems to solve)
- Game World
 - All static & dynamic objects and their behavior
 - Pickups, weapons, enemies, enemy AI...
 - List of levels/locations
- User Interface
 - All screens, UIs, HUDs
- Story, Lore, NPCs...

Core game loop



2D Camera

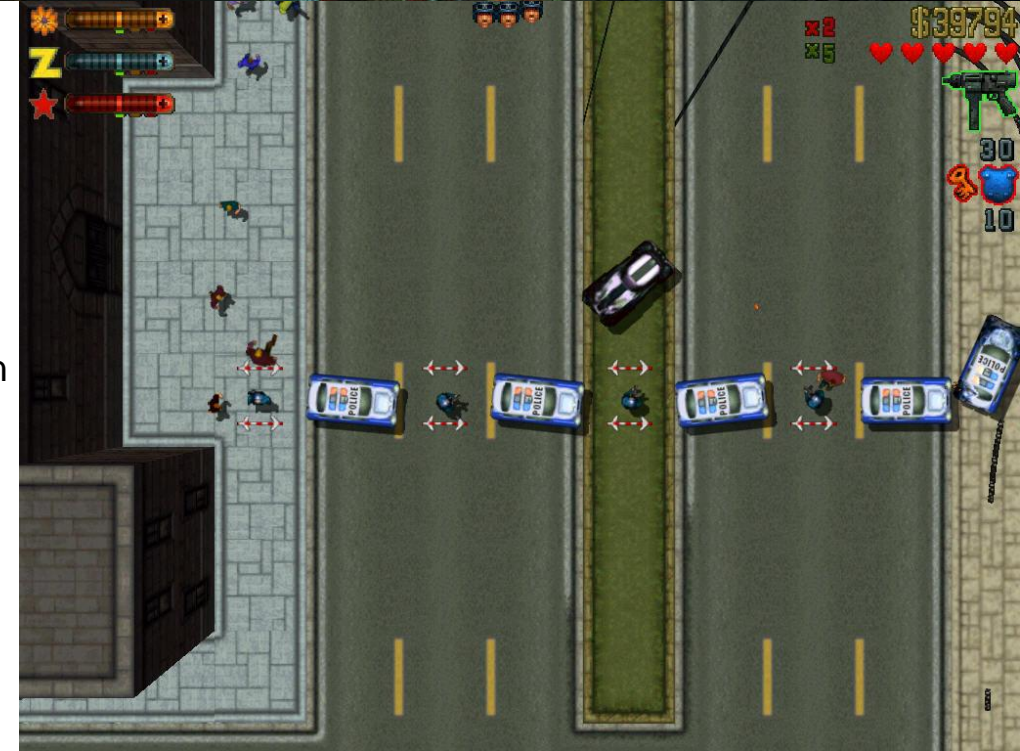
Isometric



Side-scroller



Top-down



3D Camera

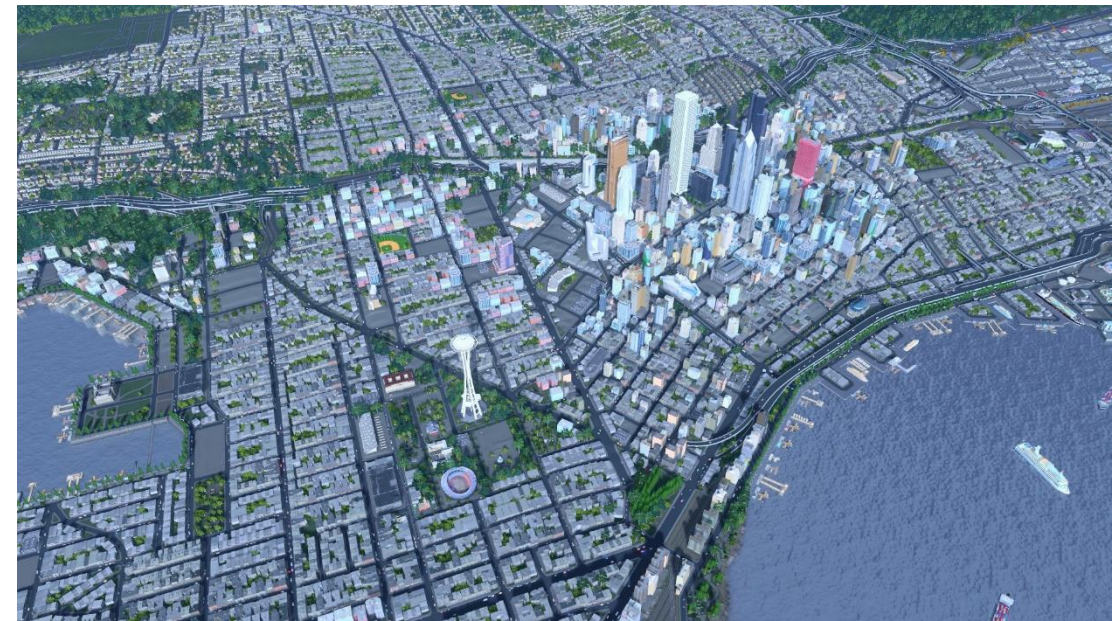
1st person



3rd person



Top-down



Controls

- PC or Console or Touchscreen or VR or Gyro or GPS or ...
- What is controllable & how



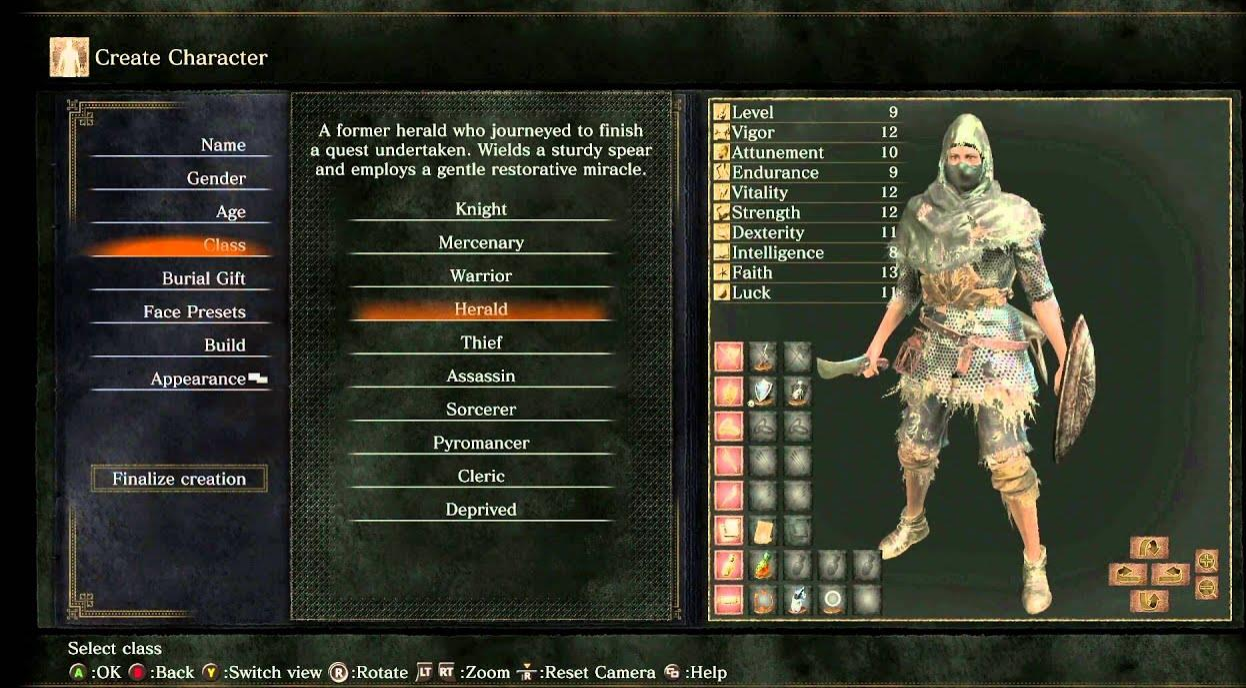
Characters

- Single character (avatar) – one person
- Multi character – party
- Lots of units – commander



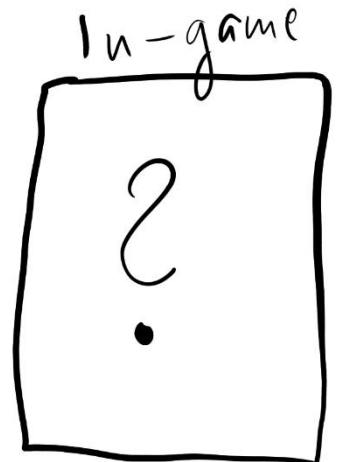
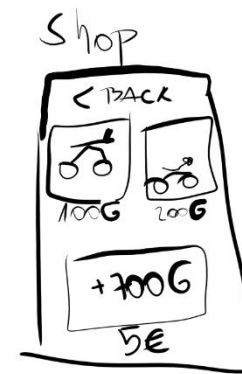
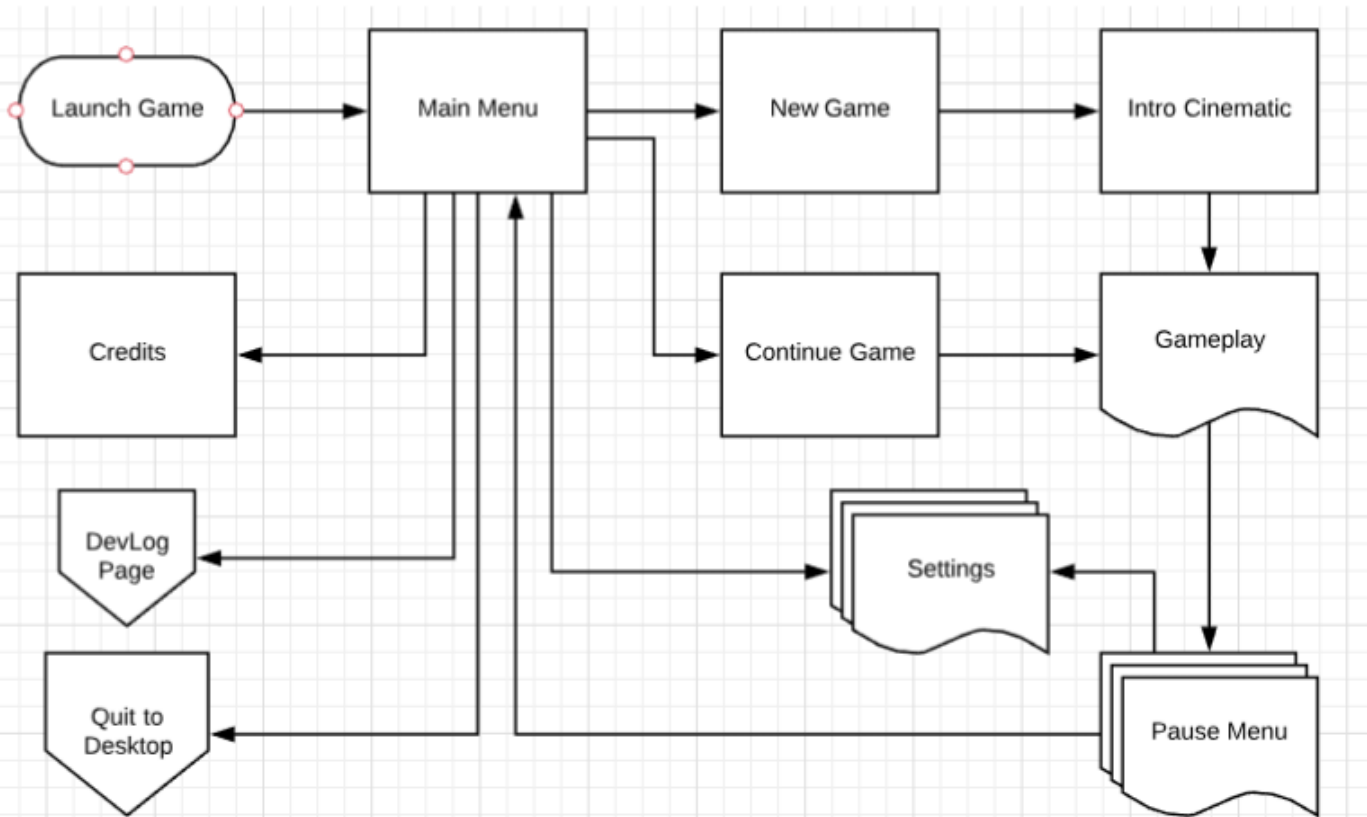
Characters

- Specific or customizable?
 - Visuals
 - Stats



UIs

SCREENFLOW



1 Picture > 1000 words

Art Design Document (ADD)

- Describe art that will be used throughout the game
- Art style, mood boards
- Color palettes
- 2D sprites, 3D models
- Animation
- Techniques, software, workflows
- Visuals & audio

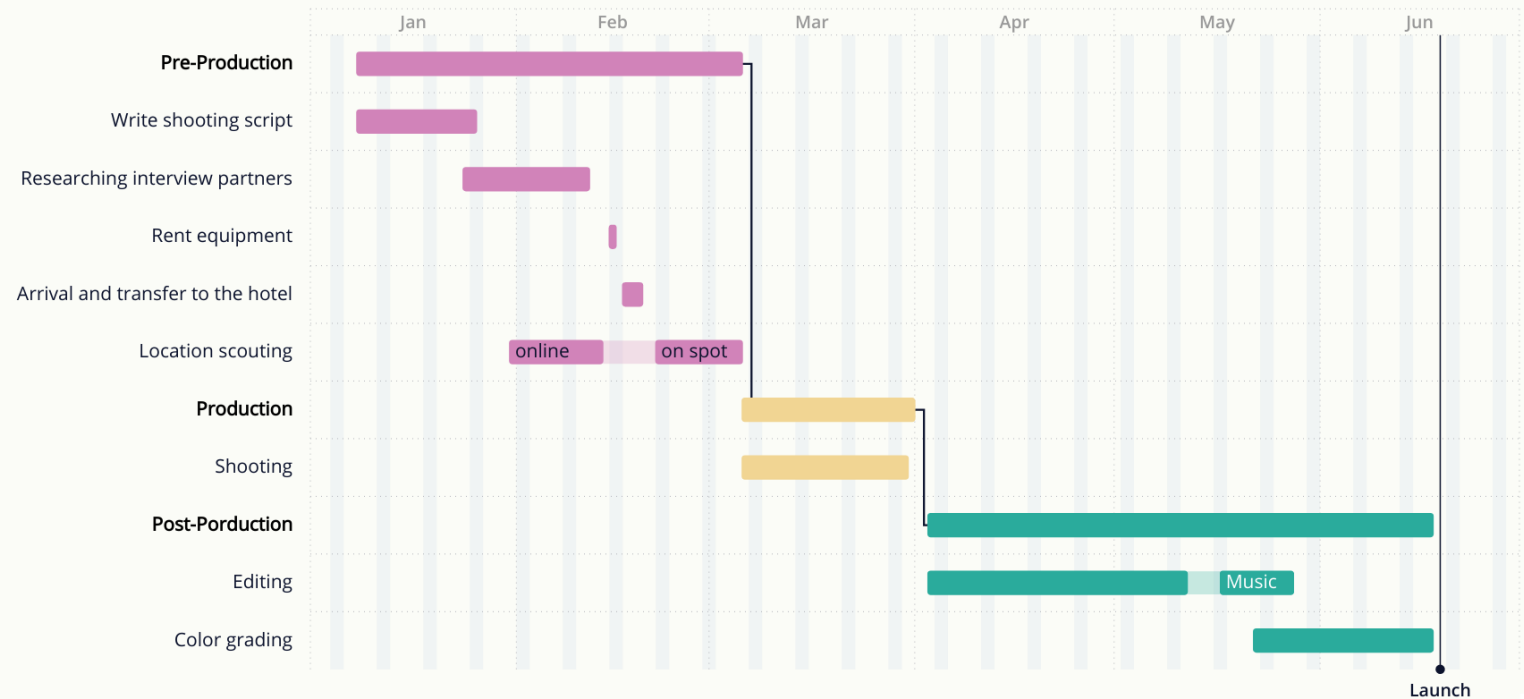
Technical Design Document (TDD)

- Describes the engine/technical solutions on which the game will run
- Comparison to other engines on the market – why did we pick our engine
- Technology production path
 - How they will get from concept to software
- Specific requirements and features, HW & SW
- Refers to the GDD/ADD
- Closer to the software requirements specification
- Farther from the concept document

Production Plan

- Required team & allocation
- Tasks & time estimates
 - Gantt chart
- When will we ship?
- Marketing strategy
- Post-launch plan
 - DLCs, LiveOps...

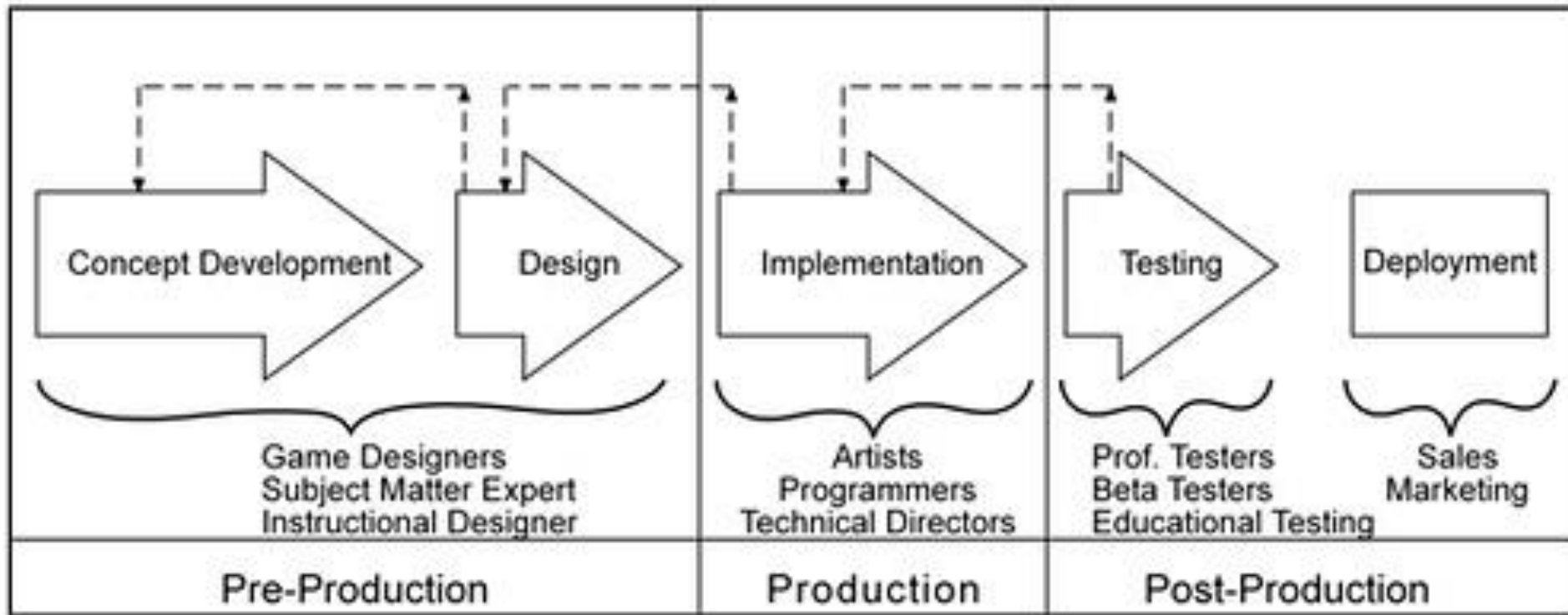
Gantt Chart Template for Film Production



Example Concept Documents and GDDs

Super Mario, Race'n'Chase, BioShock, ...

Game development stages



- Image from <https://digitalworlds.wordpress.com/2008/04/10/the-process-of-game-creation-the-game-design-document/>

Development phases

- **Concept** - Start with a small team, work on a concept document
- **Prototype** – not only gameplay, also art
 - *“A working piece of software that captures onscreen the essence of what makes your game special, what sets it apart from the rest, and what will make it successful.”*
- **Pre-production** – Create GDD, TDD, ADD, mechanics
- Production – create all content
- Alpha, Beta, Gold
- Post-production

Problem & Solutions

- Waterfall is not cool
- Moving towards Agile/Scrum
 - Very cool
- Game Development is inherently iterative
- Need to test various systems
 - **Do players like XYZ?**
 - Are the graphics good?
 - Is the music suitable?
 - Are the controls good?
- **Build-Measure-Learn**
 - Read “Agile Game Development with Scrum”

References

- http://www.gamasutra.com/view/feature/131791/the_anatomy_of_a_design_document_.php
- http://www.gamedev.net/page/resources/_/creative/game-design/tom-slopers-format-for-game-design-specifi-r243
- <http://sloperama.com/advice.html>
- <http://www.cs.tufts.edu/comp/150CIS/AnAntsLife/AnAntsLife-GameDesignDocument.pdf>
- <http://www.scribd.com/doc/53563149/Grand-Theft-Auto-Design-Document>
- <http://www.gamepitches.com/>
- <http://code.tutsplus.com/articles/effectively-organize-your-games-development-with-a-game-design-document--active-10140>

References (2)

- Novak, J. (2012). *Game Development Essentials : An Introduction*. Clifton Park, N.Y: Delmar. 3rd ed.
 - Chapters 1, 2, 3, 4, 11
 - Further reading: Chapters 10, 12
- Adams and Dormans (2012). *Game Mechanics: Advanced Game Design (Voices That Matter)*.
 - Chapter 1 – Designing Game Mechanics
- Keith, C. (2010). *Agile Game Development with Scrum*. Addison-Wesley
 - Just read the whole book!
- Schell, J. (2014). *The Art of Game Design: A Book of Lenses*. A K Peters. 2nd ed.
 - **Must read for every single game dev**