End of Pre-Production Requirements

"Pre-Production is the time when you try to figure everything out before you actually start building it!"
(Extra-Credit, The Pre-Production Problem - https://www.youtube.com/watch?v=ukADFPuscG8)

General purpose of the Pre-Production

The main objective of the pre-production is to define all goals and requirements for the production of the game and to address all potential risks and open questions and come up with solutions for them. It is the time for writing the design specs, do the overall planning and prove game mechanics and fun via prototypes and core playable.

During pre-production, you normally work only with a small group of dedicated people. This allows the creative process to be much more focused and purposeful than in production, when development is already full steam ahead with a large crew.

It is more efficient and cost effective to first specify all the requirements and the overall scope before starting the actual implementation. And it's also better for team motivation, rather than rushing into production without a clear vision and shared goals.

1. Core Playable (Proof of Fun)

At the end of the pre-production phase there must be at least one early version of the game that demonstrates the "fun". The core game mechanics and game loops should be presented in a way that clearly shows how the final game will excite and entertain players.

The creation of this build, often referred to as "Core Playable" or "First Playable", should be tackled in the first half of Pre-Production, before any major design work and documentation has started. The following essential areas related to "proof of fun" need to be covered:

- ***Core Gameloop:*** The Core Playable (as the end result of all prototypes created) demonstrates the essential key mechanics of your gameplay. It must become crystal clear what the player will spend most of their time doing in the finished game and why they will find it entertaining and enjoyable.
- ***Camera, Character, Controls (CCC):*** The commonly used acronym CCC describes the relationship between the player's input and the reactions of their "virtual representation" (such as character, vehicle, troops etc.) on the screen, including the visual staging by the game camera. In virtually every game, this interdependence has a major impact on the overall experience (and thus on the "fun"). Therefore, it should be prototyped and thoroughly tested at an early stage of pre-production.
- ***User Interface and Experience(UI/UX):*** Part of the CCC proof above also depends on how the player interacts with interfaces and menus and what feedback they receive. This could be demonstrated e.g. by click dummies, using Balsamiq, MS-Power Point or other tools – or ideally already in a rough version within the Core Playable itself.
- ***Definition of Production Value:*** Another pre-production objective is to demonstrate the desired quality of the final game in terms of graphics and audio. This can be achieved separately outside of running prototypes with the help of reference materials from other titles (video footage, benchmark graphics etc.).

2. Game Design Documentation

At the end of the pre-production phase, the development documentation – in particular the game design document – is completed to the extent that all key features and gameplay loops are defined and described.

Everyone involved is aware that a game design document is subject to constant changes and additions, and accordingly a GDD cannot have the status "final" at the end of a pre-production phase in the sense that the definitions of all values, formulas, texts and the like are completely finished.

However, the GDD must at least capture and describe the game features in such a way that all project participants can identify and evaluate the complexity of the respective work. The project plan (or prioritized backlog), which is also part of the end-of-pre-production requirements, will be based on these assumptions.

The documentation should be ideally in such a format / an environment which offers all team members rapid access and which in the case of changes, keeps everyone up to date and at the same level of knowledge e.g. through a Wiki/Intranet. Furthermore, the documentation should be as visualized as any possible (diagrams, PPT presentations, click dummies etc.). It is not about creating “epic” text documents and papers, it’s all about visualizing the game ideas and concepts.

The documentation should contain notably the following deliverable items:

- ***Game Vision:*** A description of the main goals of the game in the form of a vision statement, usually no longer than two pages. The game vision usually already exists as part of the pitching materials. However, by the end of pre-production it has been validated and finalized. At this stage, adjustments may have been made based on findings about core game mechanics and decisions about specific features. Once in production, the game vision does not change anymore. Otherwise, it automatically means that you have to start from scratch again and enter another pre-production phase to re-evaluate everything based on the changed vision.
- ***Game Design Document:*** A description of all relevant game features and core gameplay loops in a comprehensive form. The GDD should provide all project participants with a clear and as accurate as possible vision of the game and its content, enabling team members to estimate and prioritize their workload and implementation efforts (=tasks).
- ***Definition of Game Scope & Quality:*** A definition of how big the game will be (game length, size of the world etc.). In addition, publisher and developer have agreed on the means to measure the quality of the core features, e.g. by comparing them with competing products or using a quality matrix.
- ***Feature and Asset List incl. Classification:*** An overview of all the features and assets of the game. Each of the individual features has been classified as: ‘Must-have’, ‘Should-have’, or Nice-to-have’. An excellent format for such a deliverable is a product breakdown structure, also known as PBS.
- ***Art/Style Guide:*** Definition of the visual style of the game, using supplied examples as a benchmark for all graphics to be created. Both from both a technical point of view (engine, shaders, poly-count etc.) and from an artistic perspective (graphical style, concept artwork, mood boards etc.).
- ***Level-/World- & Story Guide:*** A detailed description of the game world and backstory (if any), including any scenarios, levels or settings that occur in the game. Overall scope, such as number of worlds, levels, settings (dungeon, forest etc.) and all variants.
- ***Name of the Game:*** Ideally, the final name of the game has already been determined during pre-production, legally checked, A/B tested via marketing and trademarked (including URL etc.). This is not a prerequisite, but a strong recommendation and less trivial than it may sound.
- ***Definition of Balancing Format:*** Especially but not necessarily only for Free2Play games, this definition of all relevant balancing parameters can be also required by the publisher at the end of pre-production. Such a document should define how the balancing variables are implemented in terms of format and how they can be updated once the game is live (Database, Balancing Sheet/Google Doc etc.).
- ***Monetization Strategy (if applicable):*** When creating a F2P game, a part of the Game Design Document needs to cover a clear Monetization strategy, explaining the core Monetization and Gating loops and mechanics, including:

- Description of the core systems & gameplay, economy & progress loop, gating
- Currencies: major sinks, sources (and: are these valid in all stages of the game, early to end game – if not: indicate the phasing)
- Collection of key competitors, and main top level differentiation points from those
- Outline: expected LTV by item category / feature (in USD and in %)
- Outline: High Conversion Items
- Outline: Whale Items

3. Project Planning

At the end of the pre-production stage, the team has delivered a solid planning basis for the production phase.

These project and schedule plans are created from the results and findings of the pre-production phase and, in particular, from the core playable and the game and tech design documents.

The planning usually includes the following items:

- ***Project Requirements:*** All conditions or requirements that must be met for the game to be successful were identified and their impact on time, quality and resources was assessed. The goals of both the publisher and the development studio were aligned.
- ***Project Plan/ Backlog:*** A meaningful project plan and/or task backlog has been created. All work packages, tasks or epics/user stories are identified, including assigned durations (=estimated time or complexity) as well as priorities and dependencies.
- ***Resource & Staffing Planning:*** The resource planning (personnel) was scheduled directly, deviating from the project plan. It includes a staffing plan with dates for introducing each resource to the project and a definition of each team member's role. Some publishers also require a list of key personnel that includes all essential regular or senior level team members who cannot be removed from the production without written notice.
- ***Budget Planning:*** Planning for the required development budget has been completed and reflects both the resource and staffing plan as well as the overall project plan.
- ***Business Case:*** Normally a cost-benefit analysis is done by the publisher rather than the development team. Depending on the form of collaboration however, it may make sense for the developer to also create a business case reflecting different outcome scenarios, especially if there are plans for further GAAS (games as a service) activities beyond release.
- ***Risk Analysis:*** A risk matrix has been created listing the risks currently perceived as most urgent for the project. The development team should evaluate the risk list regularly (usually: weekly) during the production phase and always update it accordingly.
- ***Milestone Definitions:*** Derived from the project plan, a coherent milestone plan has been defined. There are separate definitions for each milestone and deliverable. However, keep in mind that these goals will be revised during production. It is usually smart to define only the next 2-3 milestones in detail. All others can remain more on a meta-level and will be fleshed out later when the goals become more specific.
- ***QA-Planning:*** Planning for testing & QA has been completed, taking into account the publisher's schedules, workflows and tools for QA, as well as the development studio's requirements, including definition of bug tracker, ticket format, required cheat codes and other documentation needed for testing. These can include:
    - Definition of tester roles in collaboration with Publisher

    - Definition of Bug Tracking Tool, considering Publisher´s standard bug tracking tool in order to make cross office communication easier
    - Definition of bug template formats and other tickets considering the standards Publisher uses
    - Tools, cheats and documents required for testing (test plans) are defined in collaboration with Publisher´s QA
- ***Localization:*** The planning of translations and adaptation of the game into different languages has been scheduled. This usually includes:
    - The code needs to be localization-ready, which means it avoids hard-coded strings, the ability to display non-Western languages and different time/date/currency/number formats.
    - The in-game text needs to be delivered in a compatible format to the Publisher's localization platform
    - Remove all redundant/superfluous/outdated localizable content prior to starting the English Editing phase.
- ***Marketing Style Guide and Positioning:*** A style and positioning guide was created for the publisher's marketing team. It provides information about planned marketing assets, trailers and social media efforts. It is also intended to help the marketing team develop a deeper understanding of the product in general and its positioning.
- ***Marketing Asset Plan:*** The development studio should provide a schedule of assets to be delivered to the publisher for marketing purposes and establish a pipeline for consistent, updated delivery.
- ***Definition of Scope Soft Launch and Hard Launch (if applicable):*** For Free2Play games, a definition of the planned feature set was created for soft launch as well as for world-wide launch (aka hard launch).

4. Technology

At the end of pre-production, all major technical questions and problems are addressed and solved, also taking into account the publisher TRC's (such as integration of API's etc.).

The technical planning, documentation and, if necessary, prototypes have been created in way that a technical staff member from publisher can make a judgment about the complete technical base framework (Code Review) and is able to approve it.

The technical documentation (and proof of concept/prototypes if necessary) must essentially cover the following areas:

- ***Software Architecture Documentation:*** A software architect document showing the planned technical realization is produced at programming and architectural level in such a form that a technical person outside the project can review and assess how the game works. It takes into account all requirements from the GDD as well as technical goals such as server-client communication etc. (if applicable)
- ***Essential Toolbase:*** All the essential tools that are fundamental to the creation of the game are available and fully functional. All team members, e.g. in art or in level design, can immediately start creating content as soon as production begins.
- ***Technical Base Framework:*** The complete technical framework, especially for the engine (graphical presentation, lighting, shaders etc.), has been completed and can be assessed and approved on the basis of prototypes.

- ***Middleware Evaluation:*** The evaluation of all middleware needed for the game has been completed and the middleware can be licensed immediately after the end-of-pre-production milestone has been accepted.
- ***Platform specific Technical Issues:*** During pre-production, platform-specific technical development issues were addressed. These may include for example TRCs, handling of fonts (license, support Asian fonts etc.), transfer of the game to different OS/platforms, size limitations, definition target specifications, handling of permanent connection (for online/mobile games), performance definition, SDK's, UI handling from a technical point of view and much more.
- ***Build Pipeline:*** The team has a build pipeline in place that can produce proper builds, including nightly or weekly builds, branching etc. For work-for-hire or mobile game productions, the team may also need to demonstrate that they can align with the publisher's build pipeline and deliver builds via publisher-approved tools such as HockeyApp on mobile (using a source control such as SVN with a dedicated file structure for committing builds).
- ***Technical Risk Evaluation:*** During pre-production, the biggest risks for the technical implementation were assessed. It is clearly defined whether some of these risks still exist in this area for the actual production phase and how they should be dealt with.
- ***Source Code Dump:*** The developer provides publisher with access to either the source code of their client or sufficient code samples so that Publisher may conduct a general code review. In addition, publisher may require a complete source code dump with each milestone delivery.
- ***Publisher TRCs and APIs:*** The implementation of all APIs or tools required by the publisher have been discussed and agreed. These can be, for example, localization tools but also shop servers, monitoring or tracking tools, promotion tools, feedback servers, cloud monitoring etc. There are no unknown technical risks or open questions regarding these APIs and their implementation during Production phase.
- ***Server Definition & Requirements (for Online, Mobile/F2P):*** For games with online components, part of the software architecture documentation mentioned above must provide a detailed overview of the server architecture. Potential publisher-specific requirements such as Amazon AWS, OpsWorks, database structure requirements, etc. have been taken into account. QA requirements for test servers in the different project phases are also considered.

5. Workflows and Processes

In addition to task planning, resource allocations and budgeting there is another area that should be closely evaluated during pre-production and that is the overall organizational structure within the development studio.

It is strongly recommended that all essential workflows, communication processes and project management tools and methods are defined in consultation both internally with all team members and externally with the publisher.

In particular, the following aspects should be addressed in detail:

- ***Team Structure:*** Definition of the team structure, for example in the form of a team org chart. There should be also a schematic representation of the communication processes within the individual departments as well as a list of all important team members.
- ***Definition of Project Roles:*** A precise definition of the project role of each team member (tasks, areas of responsibility and interactions).
- ***Reporting - Internal / with Publisher:*** Definition of how and with what temporal frequency communication should take place. This should include weekly reports, daily stand-ups, weekly meetings, meeting minutes etc. (whatever is applicable).
- ***Outsourcing:*** Definition of how the cooperation with outsourcing partners should work, including responsibilities, evaluation of potential partners, communication processes, approval process for deliverables, etc.

- ***Change Requests Process/Change Control:*** (aka Change Order) Definition of how change requests will be managed throughout the project, both internally and with the publisher. How are change requests submitted and tracked? Who decides and through which process? Which change requests should be implemented?
- ***Approval Processes (internal/Studio):*** Definition of how work packages and assets are to be shared. What are the dependencies and areas of responsibility? Who approves what and by when? How should assets and work packages be shared and managed by outsourcing providers?
- ***Approval Processes (external/with Publisher):*** At what level is the publisher involved in which approvals (other than milestone approvals)? How are communication workflow and responsibilities for such approvals handled?
- ***QA Processes:*** Detailed definition of how QA should work within the development studio, taking into account the QA standards and requirements of the potential publisher (e.g. bug tracking software, QA submission process, store TRCs, tools, templates and documentation).
- ***Version Control and Back-up:*** Accurate documentation for the use of version control software. How are work packages checked in? How must they be documented (version history)? In addition, the functioning of the software backup process within the studio must be documented.
- ***Proven Production-Ready Pipelines:*** Definition of how the production pipeline works. How does an asset get into the game? Are engineers involved in this process or can, for example, a graphic asset be brought into the engine by an artist himself, without additional programming work?

CHECKLIST

Pre-Production Deliverables

- ☑ The project goes through an initiation phase in which game vision, high concept and an initial definition of the overall project goals are created before the start of pre-production.
- ☑ In addition, the desired end results and appropriate measuring criteria for success are defined with clear deliverables before the actual phase begins.
- ☑ Sufficient time is allocated for a thorough pre-production.
- ☑ The pre-production team is large enough to do its job, but still small enough to act quickly and not produce any unnecessary overhead.
- ☑ The pre-production team consists mainly of experienced senior and lead staff.
- ☑ The team does not produce unnecessary design documentation until the “fun” is proven through prototyping and a core playable.
- ☑ No work is done on any planning or Milestone documentation until the specifications (GDD, TDD) are written.
- ☑ The project does not move into production before pre-production has been properly completed and all objectives have been met. There are no shortcuts.