Einleitung

* Ausgangssituation (Platformübergreifend Spiele —>Probleme)
* Motivation (warum tun wir das was wir tun?)  
  Warum haben wir einen Sidescroller gemacht?
* Vorgehen ( Projektplan: Aufteilung, Teams, Wer welches Thema -> Plan erzeugen wer was gemacht hat)
* Gegebenfalls Struktur der Dokumentation

Grundlagen (projektspezifisch)

* Frameworks vorstellen (Warum? Vorteile was bietet es?, warum nicht anderes)
* Welche IDE mit welchem Plugins (hier erwähnen was verwendet wurde + Probleme, EVTL. ein Tutorial (4 Seiten mit Bilder) erstellen)
* Sprites (was ist das überhaupt u. wie funktioniert das im Programm? Allg. und in Cocos, Funktionen, wie verwendet man sie, Wo ? )
* Steuerung( Was bietet Engine, Was ist Möglich, Warum Unsere Steuerung)  
  Warum sind im Jump-nRun Level bei bestimmten Tätigkeiten andere Buttons disabled?
* TileMaps (was ist1 das und wie funktioniert es?)
* Scene Prinzip von Cocos2d ( Allg Videospiele, Welche Mögl., Warum Unser Scene Prinzip)
* Audio Prinzip von Cocos2d ( Komprimierung, Welche Möglichkeiten gibt es, Warum Unser Audio Design)
* Animationen Prinzip von Cocos2d( Bewegungen also Actions, SpriteAnimationen) -> Unterscheidung zwischen Bewegungen von Sprites die wie Animationen wirken und Animation von Sprites über Spritesheets und plists
* UserDefault ( Was ist das, Wie funktioniert das)

Architektur

* muss korrekt sein
* unsere Architektur
* Diagramme
* Anforderungen
* Design ( Warum Wir diesen Aufbau haben, Warum 3 level+ Bosslevel usw.)

Implementierung

* Konzepte aus Grundlagen und Architektur dem Leser näher bringen, damit man es verstehen kann (kleine Codebeispiele)
* Bug Risiken (auch konkret….)
* Laden von Tilemaps
* Stage Hazard und Projektile Coins
* RandomLevel

Evaluation (Vergleiche zu anderen Bibliotheken z.B. Unity, AndEngine)

* ein Beweis dass das was wir gemacht haben auch funktioneirt
* Bilder, Screenshots ...
* Was kann meine Software, was kann sie (noch) nicht
* UseCase Diagramm (z.B. BossLevel)

Fazit und Ausblick

* ehrlich bleiben
* Was haben wir erreicht?
* Was funktioniert?
* Was funktioniert nicht und warum?
* Was könnte man noch machen? Damit es super wird!

Anhang:

* Alles was zu viel ist. Z.B eine komplexe Klasse

Ausgangssituation:

Es wurde ein Framework vorgestellt und jetzt wollen wir sehen was wir damit erreichen können

Tutorial: dem Leser verständlich machen wie man alles einstellt damit man arbeiten kann

Sachliche Sprache!