

Final report marking sheet

Student's name: Christopher Alastair Irvine	Student's Reg: 100036248
Marker's name:	Supervisor <input type="checkbox"/> 2nd marker <input type="checkbox"/>
Title: CODEX: A Progressive Web App in React for table-top role-playing games	

	Marks
Introduction, related work and context	/20
Design / Methodology / Experimental plan	/20
Outcome / Analysis / Results	/30
Discussion, evaluation and conclusion	/20
Quality and accuracy of writing	/10
Overall mark	/100

Comments

Signed:	Date:
---------	-------

Presentation marking sheet

Student's name: Christopher Alastair Irvine	Student's Reg: 100036248
Marker's name:	Supervisor <input type="checkbox"/> 2nd marker <input type="checkbox"/>
Title: CODEX: A Progressive Web App in React for table-top role-playing games	

	Marks
Explanation of problem and solution	/10
Software design / analysis of investigation / results of experiments	/30
Clarity of slides	/10
Structure of presentation	/10
Oral performance, including timeliness	/10
Evidence of understanding and reflection	/20
Quality of supporting flyer	/10
Overall mark	/100

Comments

--

Signed:

Date:

Student's name: Christopher Alastair Irvine	Student's Reg: 100036248		
Title:			
Supervisor: Dr Katharina Huber	Signed:	Date:	
2nd marker:	Signed:	Date:	
Marks	Supervisor	Second marker	Agreed mark
Report			
Presentation:			

Comments (made available to the student)

Christopher Alastair Irvine

Registration number 100036248

2018

CODEX: A Progressive Web App in React for table-top role-playing games

Supervised by Dr Katharina Huber



University of East Anglia
Faculty of Science
School of Computing Sciences

Abstract

This project is the tits and here's why...

Acknowledgements

I would like to thank Dr Katharina Huber for taking on the supervision of this project, and guiding me towards success. Additionally I would like to thank Wizards of the Coast for their generosity and kindness in allowing the use of their Intellectual Property for this project.

Contents

1. Introduction	6
2. What is CODEX?	6
2.1. Context	6
2.1.1. What is Dungeons and Dragons?	6
2.1.2. How does a Dungeon Master differ from a Player?	9
2.1.3. Why was the CODEX app developed?	10
3. Related Work	11
3.1. Dungeon and Dragons Apps	11
3.1.1. D&D Beyond	11
3.1.2. Roll20	11
3.1.3. ProD&D Dungeon Generator	11
3.2. Software Engineering	11
3.2.1. What is Software Engineering?	12
3.2.2. What is Agile?	12
3.2.3. Agile Solo	12
3.2.4. XP for One	12
3.3. Web App Technology	12
3.3.1. What is a Web App?	13
3.3.2. ReactJS	13
3.3.3. Semantic UI	13
3.3.4. Databases within Web Apps	13
4. Development and Implementation	13
4.1. Using Agile Solo	13
4.2. Design	13
4.3. Development Observations	14
5. Outcome of the CODEX project	14
5.1. Development of CODEX	14

5.2. Effectiveness of Agile Solo	14
5.3. Feedback on the CODEX app	14
6. Evaluation of CODEX	14
6.1. Development Issues	14
6.2. Agile Solo Evaluation	14
7. Conclusions	15
7.1. Agile Solo	15
7.2. CODEX	16
A. CODEX Gantt Chart	17
B. Worked Example of Dungeons and Dragons	17
References	19

List of Figures

1. The above two figures show the typical equipment for a D&D game 8
2. CODEX Gantt Chart, outlining the major tasks and deliverables 18

1. Introduction

hello there

2. What is CODEX?

CODEX was a project that has two components. The first, produce a progressive web app built in ReactJS that was developed using a single developer Agile software engineering methodology, the second component was to evaluate the quality of that methodology. By the end of this section we will have an appreciation of the CODEX app, in terms of both context and purpose (see Sections 2.1 and ?? respectively).

2.1. Context

The CODEX an app was based around the popular tabletop role-playing game known as Dungeons and Dragons (D&D), created and published by Wizards of the Coast. This game, and the principles that bring the game to life, drove the requirements which formed the basic structure of the CODEX app. Therefore, in order to understand the functionality, principle and importance of the CODEX app, we must first gain an understanding of D&D.

The author of this paper has multiple years of experience with D&D, both as a *Player* and *Dungeon Master* (DM). The CODEX app began as a personal project for the author of this paper, which grew into a tool to evaluate a single developer Agile software engineering methodology.

2.1.1. What is Dungeons and Dragons?

The game of D&D, at time of writing, was a popular tabletop role-playing game where a group of friends engage in a grand fantasy adventure powered by a narrative (Gygax and Arneson (1974), Peterson (2017)). The story would be divided into manageable blocks of time known as *sessions*. Each session may take between two and four hours to run, however this is not strictly enforced and is decided by the group of players. The events of each session, and the larger campaign, was created and controlled by a player

known as the *Dungeon Master* (DM) (for more details on DMs please see Section 2.1.2). However, the campaign would not always occur as written. The players, who were clueless as to the next plot in the campaign, might take the story in an entirely different direction several times a session. The DM would have to react to these unexpected inputs and redraw the narrative during each session, or attempt to guide the players subtly back on track (Mearls and Crawford, 2014b).

Whilst it was possible for DMs to purchase pre-written campaigns, published either by experienced DMs or Wizards of the Coast, many DMs would take it upon themselves to create their own worlds in which the campaign would occur or alter the pre-written campaigns to suit the needs of the group. For every game of D&D was inherently different from each other, including those following the same campaign book. Because of this difference it was hard to estimate how many sessions a campaign might take.

The players would form an entity known as the *party*, that would consist of the characters that the players embodied for the duration of the campaign. Each player controls one character, known as the *player character*, within the party. The party would serve as the groups device in how they interact with the world, this is true for the DM who would not embody one character, but every other character within the world. These characters would be referred to ask *non-player characters* and would include both allies and enemies of the party. For that reason, the external perception of the D&D might be that the players were against the DM. Nothing could be further from the truth, as it was the relationship that existed between the players and the DM that would shape the outcome of the campaign (Ewalt, 2014).

Despite the complicated nature of D&D the equipment that was needed to play the game was minimal. A group of players would only have needed a single set of die, a copy of the core books (*Dungeon Master's Guide* (Mearls and Crawford, 2014a), *Players Handbook* and *Monster Manual* (Perkins, 2014)), a set of character sheets (1 per player), pens and paper. Everything else, such as more dice, a *Dungeon Master Screen*, Campaign Books, Battle Mats and Tokens were supplementary. The die that was used for D&D are as follows; twenty-sided, twelve-sided, two ten-sided (one in increments of ten, the other in increments of one), eight-sided, six-sided and four-sided. Figure 1a is an example of what a D&D table might have looked like, whereas Figure



(a) A typical D&D set up. From left to right:

DM note pad on top of a couple of official Wizards of the Coast core books (*Monster Manual* shown), set of die, battle mat and tokens, a player character sheet, player's set of die and the Wizards of the Coast *Players Handbook*



(b) An example set of die that was needed to play D&D. From left to right: 4-sided, 6-sided, 8-sided, 10-sided, 12-sided, 20-sided and a 10-sided die in increments of 10s

Figure 1: The above two figures show the typical equipment for a D&D game

1b is an example of the different die used for D&D¹.

Throughout the campaign the party would have to face many challenges designed by the DM. These challenges might be full scale battles involving multiple powerful enemies and allies, to small scale skirmishes between only a handful of characters. Alternatively challenges could be a battle of the wills where the party have to negotiate themselves out of a tight situation with the law to passing through a magical trap set by an ancient being in order to acquire an item of treasure. The effectiveness in which the party would handle these situations are decided through a simple mathematical system.

Every character within a D&D world had a set of *attributes* that would dictate how that character interacted with the environment around them. These attributes are; *strength*, *dexterity*, *constitution*, *intelligence*, *wisdom* and *charisma* (more details of these at-

¹When discussing these die further in this paper, they will be referred to as dX where X is the number of sides the die has (for example the 20 sided die is referred to as d20). The exception to this is the secondary d10 (which has increments of 10) which is referred to as the d100.

tributes are available in Appendix B). These attributes were rated between 1 and 20, with 20 being the highest a character could achieve without magical means. The score would be divided by 10, rounding up to the nearest whole number, creating an *ability modifier* which would be applied to the role a dice (this number could be negative) If a character was highly skilled in a particular task, then a *proficiency bonus* would also be applied. This calculation is summarised in the below equation 1.

$$\text{skill check} = d20 + \text{ability modifier} + \text{proficiency bonus} \quad (1)$$

These skill checks allowed the players to interact with the world, via the player character, in a quantitative manner. The higher a skill check was the better that character performed in that challenge. Challenges would have a *difficulty class*, an arbitrary number which indicates the level of skill (or luck in some cases) needed to perform that task. If the challenge was to strike an enemy, it would be easier to wound someone wearing no armour than it was to wound a skilled warrior in plate armour.

Winning a game of D&D was no simple manner either, as there were no set win conditions. It depended entirely on the campaign that was being ran by the DM. A typical example might be that the party were a group of loyalists who were tasked with overthrowing a usurper and restoring the rightful monarch to the throne. That party would *win* where they to do so, however the objectives might change where the party to learn of some new, unsavoury, information about their beloved monarch. A campaign might end without achieving the original goal, or shorter campaigns may end with the start of a new story for the party. It all depended on the attitude of the players towards the game.

For a short, worked example of D&D please refer to Appendix B, where a scenario is explored and explained.

2.1.2. How does a Dungeon Master differ from a Player?

In Section 2.1.1, we explored some of the principles that gave D&D life, however the most important principle to the CODEX app was the difference between a DMs and Players. As briefly explained above, a DM was a member of the group who would control the narrative of the campaign and dictate what was occurring during each ses-

sion. Whilst the DM may write the campaign narrative themselves and would have had a great deal of control over it, they would not own the story. That belongs to the group as a whole. Player input during the sessions shaped the world and story as the party progressed through the campaign, forming relationships with characters that the DM portrayed.

It was the responsibility of a DM that the party was enjoying the campaign, narrative and challenges. As a result the DM would have had to have known every intricacy of the world at any given time, or be able to improvise when they did not know the answer. Only a DM could bring a world to life. Therefore the amount of information that a DM would have had to deal with would have been immense. Every DM would have a folder (either physical or digital) full of notes on places, people and events that resided within the DM's world. This folder would not only contain the events of the past, but plans for the future. Skilled DMs would rely on their folder more than any other resource during a session to ensure the correct reaction to party input. The maintenance of a DM folder was costly in both time and effort (Mercer, 2016). Whilst there was no definitive time spent in order to prepare for a session, DMs could easily spend over ten hours a week planning their D&D campaign (Holmes, 1980).

However, DMs were part of the game just like the players were. D&D was never the party against the DM, instead it was a symbiotic relationship where they would explore the world together and share the experience. Whilst a DM knew what was planned to happen next, it was decided by the players if what actually occurred.

2.1.3. Why was the CODEX app developed?

As mentioned in Section 2.1, CODEX began as a personal project for the author of this paper. The original functionality of the app was to be a platform for tracking combat during games of D&D. Since then the functionality has grown to include a note taking system and several other minor features that will be discussed in Section 4. Whilst there are several other systems (explored Section 3.1) that assist a DM, they often only perform one task. The CODEX app aimed to be the assistance app for DMs by possessing a wide range of functionality.

DMs spend a lot of time and effort in running their games of D&D (as discussed in

Section 2.1.2). The CODEX app attempted to remove some of the more laborious tasks from DMs and put more enjoyment into the game by removing some of the pressure. As of the time of writing, the principle functionality the CODEX app was to track combat and provide an interactive database of information about the world for quick reference.

3. Related Work

In this Section we will explore the work that was pertinent to the CODEX project. We will have a greater appreciation of the CODEX app, by evaluating the similar systems available at the time of writing, and gain an understanding of the Software Engineering discipline in addition to the Web-App architecture through the review of academic works.

3.1. Dungeon and Dragons Apps

At the time writing, there were several applications that existed to assist in the running of D&D. These apps had varying levels of functionality, however they differed from the CODEX app. For example, the official D&D Beyond app (Wizards of the Coast and Curse, Inc., 2018) (discussed further in Section 3.1.1), had a huge range of functionality including digital character sheets and a quick reference database. This service is aimed towards the players of the game, not towards the DMs, therefore D&D Beyond does not support session planning and encounter generation to the same level as the CODEX app.

3.1.1. D&D Beyond

3.1.2. Roll20

3.1.3. ProD&D Dungeon Generator

3.2. Software Engineering

The CODEX project was not only the development of a web-app based around the D&D game. It was the evaluation and examination of an agile single developer software

engineering methodology. In order to perform this evaluation, we must first have an understanding for the Software Engineering Discipline (see Section 3.2.1). From there we will learn about the Agile principles and how they changed Software Engineering (see Section 3.2.2). Finally, we will examine two potential single developer agile methodologies which would be compatible with the development of the CODEX app (see Sections 3.2.3 and 3.2.4).

3.2.1. What is Software Engineering?

boo

3.2.2. What is Agile?

boo

3.2.3. Agile Solo

boo

3.2.4. XP for One

boo

3.3. Web App Technology

The CODEX app was designed to be a Web Application (web-app). In this Section we will learn what a web-app is and how it differs from the traditional website and application (see Section 3.3.1). ReactJS is a popular library for the JavaScript programming language that was developed specifically to develop web-apps. Due to the popularity and availability of supplementary materials of ReactJS, it was selected to be the principle programming language for the CODEXApp (see Section 3.3.2). Semantic UI (User Interface) is a package developed for ReactJS systems, which would allow developers to quickly create professional UI for web-apps and was used in the development of the

CODEX app (see Section 3.3.3). Finally in this Section we will look at how Databases would operate within web-apps (see Section 3.3.4).

3.3.1. What is a Web App?

Web-apps were differentiated from the traditional website in that they could be easily ported onto mobile devices as a separated application on that device. Which can still be accessed through web addresses.

3.3.2. ReactJS

boo

3.3.3. Semantic UI

boo

3.3.4. Databases within Web Apps

boo

4. Development and Implementation

boo

4.1. Using Agile Solo

boo

4.2. Design

boo

4.3. Development Observations

boo

5. Outcome of the CODEX project

boo

5.1. Development of CODEX

boo

5.2. Effectiveness of Agile Solo

boo

5.3. Feedback on the CODEX app

boo

6. Evaluation of CODEX

boo

6.1. Development Issues

boo

6.2. Agile Solo Evaluation

boo

7. Conclusions

The CODEX project was to software engineer a progressive web app built in ReactJS, using an agile methodology. The principle challenge of CODEX was that, unlike the majority of software engineering project, there was only one developer. Agile methodologies are designed to be used by a group or groups of developers, with designated roles for individuals within the team. As part of the preparation for CODEX, a single developer methodology had to be found, these were *Agile Solo* and *XP for One*. Agile Solo was selected to be the principle methodology for the development of CODEX.

7.1. Agile Solo

Agile Solo, as described in Section 3.2.3, is a methodology that was developed because there was no Agile development methodology designed for solo developer projects. The purpose of a software engineering methodology is to provide a platform to assist with time and task management for a project, in that respect we need to have a certain amount of discipline from the developer when applying a methodology to a project. [It was this discipline that came a critical issue with a solo developer project. When working in a team, each developer is held accountable by the rest of the team. But in solo developer projects no one is holding the developer to account, with the notable exception of a supervisor when present.] However, Agile Solo functioned without flaw when implemented as the original author described. This is evidenced by the project journal, which can be found in the CODEX portfolio, where there are numerous entries that prove the following of Agile Solo. The work that was allocated each cycle was achieved and any issues in time spent of that work was down to an incorrect estimation during the allocation process.

In summary, Agile Solo has proven to be a highly effective and accurate methodology for solo developer projects. The methodology proposed in 2011 by Nystöm does not need any alterations as far as CODEX is concerned.

7.2. CODEX

- The Codex app was a large ambitious project with many complicated features
- Developed in a new language to the developer
- Developer achieved a working system that provided the core functionality of the app that met the requirements
- Many advanced features are easily integrated with more work
- Project was not a success as all features were not completed, however it was not a failure as the requirements were satisfied
- Therefore the CODEX app is a challenged project in accordance with the Standish Group Chaos Report classifications

A. CODEX Gantt Chart

B. Worked Example of Dungeons and Dragons

Strength dictates how strong a character is and would be used in determining physical challenges of raw power. Dexterity indicates the precision a character would move with, and can be used for sneaking and dodging actions. Constitution would reflect the tenacity of a character, this attribute dictates the number of *life points* a character would have and how resistant to disease he/she is. Intelligence can be used to investigate a crime scene and would typically indicate how smart a character is. Characters with a high wisdom attribute would notice things quickly and are more in tune with the spiritual. Finally, the Charisma attribute would dictate how sociable a character might be, determining that characters effectiveness at persuasion and deception.

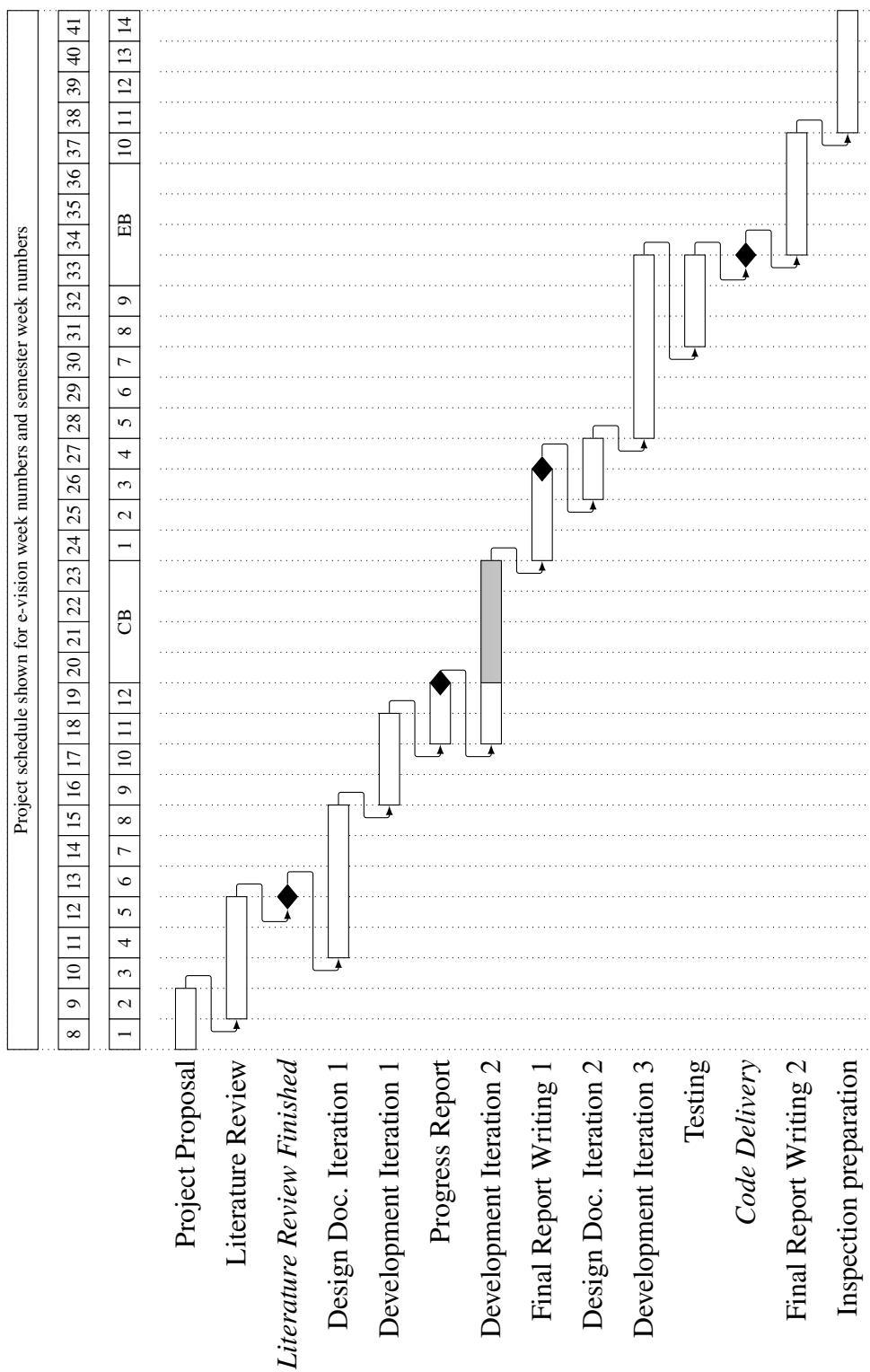


Figure 2: CODEX Gantt Chart, outlining the major tasks and deliverables

References

- Ewalt, D. M. (2014). *Of dice and men: The story of Dungeons & Dragons and the people who play it*. Simon and Schuster.
- Gygax, G. and Arneson, D. (1974). *Dungeons and dragons*, volume 19. Tactical Studies Rules Lake Geneva, WI.
- Holmes, J. E. (1980). Confessions of a dungeon master. *Psychology Today*, 14(6):84–94.
- Mearls, M. and Crawford, J. (2014a). *Dungeon Master's Guide*. Wizards of the Coast.
- Mearls, M. and Crawford, J. (2014b). *Player's Handbook*. Wizards of the Coast.
- Mercer, M. (2016). Setting up your gamemaster's screen! (gm tips w/ matt mercer). <https://www.youtube.com/watch?v=YRMV7mbe-ls>.
- Perkins, C. (2014). *Monster Manual*. Wizards of the Coast.
- Peterson, J. (2017). Fourty years of adventure. <http://dnd.wizards.com/dungeons-and-dragons/what-dd/history/history-forty-years-adventure>.
- Wizards of the Coast and Curse, Inc. (2018). D&d beyond. <https://www.dndbeyond.com>.