Cribbage Pegboard Mobile App

Aime Celeste Thayer

Western Governors University

**Table of Contents**

[Proposal Overview 3](#_Toc441469365)

[Problem Summary 3](#_Toc441469366)

[IT Solution 3](#_Toc441469367)

[Implementation Plan 3](#_Toc441469368)

[Review of Other Work 3](#_Toc441469369)

[Relation of Artifacts to Project Development 4](#_Toc441469370)

[Project Rationale 4](#_Toc441469371)

[Current Project Environment 4](#_Toc441469372)

[Methodology 4](#_Toc441469373)

[Project Goals, Objectives, and Deliverables 5](#_Toc441469374)

[Goals, Objectives, and Deliverables Table 5](#_Toc441469375)

[Goals, Objectives, and Deliverables Descriptions 6](#_Toc441469376)

[Project Timeline with Milestones 6](#_Toc441469377)

[Outcome 7](#_Toc441469378)

[References 8](#_Toc441469379)

[Appendix A: 9](#_Toc441469380)

[Appendix B: 10](#_Toc441469381)

[Appendix C: 11](#_Toc441469382)

[Appendix D: 12](#_Toc441469383)

# Proposal Overview

The cribbage pegboard mobile app allows a user to carry around only two items (their phone + playing cards) where before, enthusiasts had to carry around three items (playing cards + a cribbage pegboard, while presumably their phone always goes with them everywhere). The mobile application also saves users’ win record and scores (either 121 for wins, or the exact score value =<120 if lost) to enable betting/settling wagers based on relative scores. People enjoy the speed of the game, the chance to use basic math skills, and potentially the competitive aspects.

## Problem Summary

Smartphones are great – there are many captivating games out there for people who have a phone in the late 2010s. However, some people prefer something tangible, like playing cards. They also might like card games like cribbage, which have an accessory score board. Because of this, we’ve chosen to develop a cribbage score app for smartphones. The cribbage board is a bit bulky and annoying to carry around, but because of their convenience and feature set, a large percentage of adults carry around a phone. Some subset of those people also carry a deck of cards but don’t wish to have a large cribbage board with them everywhere they go. The app we are developing is meant to satisfy those users.

## IT Solution

The proposed IT solution for lightweight carry, maximal feature, on-the-go gaming, is for users to carry around a phone and a deck of cards. On the phone, users would have a virtual cribbage score board app to keep track of their game tallies, and would allow for multiple user save locations and multiple user names (i.e. one user profile could have its own custom user name). This app would have more than one game board style (though they all add up to 121 points) that the user could choose at the beginning of a new game. Users who have used the app before can clear their old scores, or maintain a running tally of their games.

## Implementation Plan

The implementation plan for this software application shall occur in 4 phases.

* Phase 1: Program estimation
  + This phase includes:
    - creating storyboards
    - making an architecture diagram
    - outlining discrete tasks in each phase
* Phase 2: Wireframes and artwork
  + This phase includes:
    - creating digital images of cribbage pegboards
    - identifying menus and menu items
    - making buttons and other icons
* Phase 3: Software implementation
  + This phase includes:
    - the UI hookup to code
    - creation of general program functionality
* Phase 4: Quality assurance
  + This phase includes:
    - checking that the delivered software matches its original specs
    - that functionality is as expected / designed

# Review of Other Work

When reviewing relevant related works, we reviewed four online articles from both academic / educational and news sources. These primarily had to do with the rules and experience of playing cribbage, as there are few online or mobile cribbage apps, and no related white papers or research studies (at least, not that we were able to find).

Using these resources, we settled on an authoritative rule set and developed guidelines to attempt to craft an experience that is equivalent to those described online. Some players play cribbage competitively, and we wanted to develop a mobile app which was suitable for both casual players and more-serious competitors. For example, one user stated “I like the speed of the game and having to make decisions quickly…With all the adding and counting, it helps keep you sharp.” Another user in that same article stated “You can’t predict the outcome…Things can change in a flash. You might be at the end of a game and need two points to win while your opponent needs 22 and he can end up beating you. It’s not over until it’s over.” (Klemenc, 2013)

## Relation of Artifacts to Project Development

In “Cribbage: it’s not just a game, it’s an obsession”, we found anecdotes explaining why users choose to play cribbage and information about how cribbage tournaments might be run. We also learned that a wide variety of people (i.e. different ages, professions, and relationships to one another) play cribbage. These users also play cribbage frequently – some as often as daily. We found this an encouraging reason to develop a cribbage app – since it’s only a fun game if people play it!

When reading about tournaments in this same article, we learned, “small local tournaments where players typically are seated at a long table and play nine games with different opponents, with play normally taking two to three hours. Players ante up $7 a night to play, with $5 going into the winners’ kitty and $2 toward expenses and various grass root prizes, plaques, certificates, jackets, clocks and trophies, which are awarded throughout the year depending on various accomplishments. … Winners also may accumulate grass roots points that help their ACC standing.” (Klemenc, 2013) which may offer interesting aspects for future releases of the cribbage app! At first we’ll replicate the scoring only, but in the future we might add relevant tutorials or tournament gameplay aspects.

In the “Rules of Cribbage”, we learned the official rules of cribbage, including the mechanics of play, how to shuffle and cut the cards, how to deal, where to put the crib cards, choosing a starter card, play order and rules of play, how to peg one’s points, how to count, and many other details of the game mechanics – all of this information either directly informs the gameplay of cribbage in our app, or the help text / tutorial we could choose to include in the app or future versions of it. (The American Cribbage Congress, 2018)

In “Amusing Cribbage Facts”, we learned some of the statistics relevant to cribbage play, details on how to call out one’s score, and the relative percentage likelihood of drawing certain card combinations. We also learned that scoring a Jack in-hand is called “his nobs”, but not the origin of the phrase (unfortunately). (Lumetta, n. d.)

Lastly, in the brief article “Facts about cribbage”, we learned about some of the discrepancies of gameplay, e.g. “The first dealer in a game wins 55% to 60% of the time.” (Cribbage Corner, 2010) We also learned more of the language or jargon of playing cribbage, like “There are several 'impossible totals' - point counts which cannot be made with any hand. The lowest such total is 19 - hence the expression 'a nineteen hand', or 'I have nineteen', almost universally (perhaps sarcastically) used to describe a zero-point hand.” (Cribbage Corner, 2010) but perhaps most interesting was the link out to the ***5 scores 2 conjecture*** which demonstrated that any hand containing a 5 card would be worth at least two points, including a Perl program which the interested reader could use to prove that conjecture by simulating a wide variety of hands.

# Project Rationale

The rationale behind this project, the interesting part of the matter, is that cribbage is a fast and lightly challenging game which allows users to have a variety of gameplay experiences. From serving up tournament play, to on-the-go “just killing time” play styles, we found value in building an app that would replace the physical cribbage board. This allows users to play wherever they’d like, as long as they have a deck of cards in addition to their mobile phone.

Technically, there’s not a lot of reason to choose cribbage over any other card game. It’s more a matter of recognizing that a smart phone is more portable and more universally useful than a cribbage board is. And since users carry phones with them most places they go these days, it made sense to replace the physical cribbage board with a mobile phone app. This allows users, some of whom play daily regardless, to practice wherever and whenever they may find themselves in need or wish of playing a game.

From a business point of view, there are few ways to monetize this app at first – besides ads. However, in future iterations, we can add features for tournament play, including a money-exchange service that could take a percentage of the winnings off the top. We could also allow users to purchase modules, everything from the aforementioned tournament widget, to tutorials and skill-building mini-games. This could also make the app more fun overall.

In summary, this app is a good way for our tech team to build skill with mobile applications, to develop a fun game that can be played anywhere, and may offer future opportunities for monetization from sales of additional mini games, modules, and tournament-style play. Of course, it will be convenient and fun for players to use – if not, there’s no reason for them to continue interacting with the game.

# Current Project Environment

We have a development team consisting of one student. Thus, we find it challenging to do much parallelization in order to speed the program up. However, we can do the simple and straightforward method of clearly outlining the project steps, and then seeing what tasks form the critical path. From there, we keep chugging along.

In order to prepare our student for a career in technology in the future, we will do a form of Agile development, with a “backlog” consisting of all the development steps, and the developer will estimate each task before scheduling it. From there, our student will tackle the task, see how close the estimation was, and then pick up a subsequent task from there. At the end of the program development cycle, our student will also do the QA and acceptance testing, comparing what they developed to the expected result.

Because it’s a one-person development team, we struggle to describe much more detail. There are no pre-existing systems and the status is reported / reportable by a single person.

# Methodology

In order to develop a cribbage application, we chose a form of Agile development, where the developer will curate a backlog of tasks and tackle them each in order. The student developer will estimate each task before taking it on. We will iterate, seeing how close the original estimation was to the total task time required, and then pick up the next task for completion.

If it makes sense at the time, we may do a sprint retrospective at the end of every week and see what was working or not. Even if it’s only a ten minute reflection, this may prove to be useful for the developer.

# 

# Project Goals, Objectives, and Deliverables

We have four goals in creating this cribbage mobile application. The first is to successfully estimate the project; the second is to develop a set of wireframes / UI mockups that will help the developer organize their thoughts; third is to implement the software functionality and hook the backend database up to the UI elements; and finally the developer needs to QA the final project to ensure the original goals were met by the work that they produced.

## Goals, Objectives, and Deliverables Table

Every project has one or more goals. Each goal is supported by at least one project objective. Each objective is enabled by at least one project deliverable. Fill out this or a similarly organized table:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Goal** | **Supporting objectives** | **Deliverables enabling the project objectives** |
| 1 | Project Estimation (create written artifacts) | 1.a. Written artifacts: Project estimate / schedule | 1.a.i. Create a task breakdown |
| 1.a.ii. Estimate the task durations and order |
| 1.a.iii. Outline the schedule and order tasks to create a full project schedule |
| 1.b. Written artifacts: Formal report | 1.b.i. Create a report that outlines the goals of the project and deliverables (including the schedule from 1.a.) |
| 1.b.ii. Proofread and edit the report |
| 1.b.iii. Publish the report on WGU website and personal wiki |
| 1.c. Written artifacts: User guide / documentation | 1.c.i. Create the user guide / user documentation for the app |
| 2 | Wireframes / UI | 2.a. Create wireframes for the various screens of the app | 2.a.i. Create main menu wireframe |
| 2.a.ii. Create user scores wireframe |
| 2.a.iii. Create game board wireframe |
| 2.b. Create image artifacts for the cribbage board(s) included in the app | 2.b.i. Create standard “racetrack” cribbage board image |
| 2.b.ii. Create “29” cribbage board |
| 2.b.iii. Create custom board especially for this application |
| 2. c. Create icons for the application | 2.c.i. Create home icon |
| 2.c.ii. Create menu icons (e.g. save, new game, help) |
| 3 | Software implementation | 3.a. Create user / scores database in MySQL | 3.a.i. Create users table with user ID and each users’ preferences |
| 3.a.ii. Create games table, including user IDs and scores for each game (i.e. game records / history) |
| 3.b. Create program functionality in Java and SQL | 3.b.i. New game |
| 3.b.ii. Save game |
| 3.b.iii. Delete game |
| 3.c. Hook up actions to UI elements | 3.c.i. Create new game -> new icon / menu item |
| 3.c.ii. Save a game -> save icon |
| 3.c.iii. Delete game -> delete icon |
| 3.c.iiii. Set preferences per user |
| 4 | QA | 4.a. Compare the application to its original goals | 4.a.i. Test each piece of functionality from 3.b. |
| 4.a.ii. Write an additional section, “Final Results” of the report from 1.b. detailing the results of the test from 4.a.i. |

## 

## Goals, Objectives, and Deliverables Descriptions

The first project goal is to estimate the project and create a set of written artifacts that support the future app development. The written artifacts include the schedule, including all tasks and their proper order; a formal report that codifies the schedule and deliverables; and a user guide that documents the program functionality and its use.

The second project goal is to create the user interface wireframes and all image artifacts, such as the cribbage boards and any icons found within the app and its menus, as well as the main home icon found on the Android apps list on the user’s mobile device.

The third project goal is to implement the software itself, including creating a DB and its schema and hook up all software functionality to the relevant image artifacts.

The last project goal is to QA that the project behaves as designed, and to write up those findings in an addendum to the project report from goal #1.

# Project Timeline with Milestones

Below, you will find a projected timeline table, with milestones for the project including the duration and start and end dates of each milestone.

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone or deliverable** | **Duration**  **(hours or days)** | **Projected start date** | **Anticipated end date** |
| Project schedule | 2 days | 8/1/2018 | 8/3/2018 |
| Formal report | 4 days | 8/4/2018 | 8/10/2018 |
| User guide | 2 days | 8/10/2018 | 8/12/2018 |
| Wireframes | 2 days | 8/14/2018 | 8/16/2018 |
| Image artifacts | 3 days | 8/17/2018 | 8/21/2018 |
| Icons | 1 day | 8/22/2018 | 8/22/218 |
| Software app | 10 days | 8/23/2018 | 9/5/2018 |
| QA testing | 9 days | 9/6/2018 | 9/17/2018 |
| QA report updates | 3 days | 9/18/2018 | 9/20/2018 |

# 

# Outcome

The cribbage pegboard mobile app allows a user to carry around only two items (their phone + playing cards) where before, enthusiasts had to carry around three items (playing cards + a cribbage pegboard, while presumably their phone always goes with them everywhere). The mobile application also saves users’ win record and scores (either 121 for wins, or the exact score value =<120 if lost) to enable betting/settling wagers based on relative scores. This project will be a success once we have a cribbage application, supporting user guide/documentation, and final report (inclusive of the results of QA testing). This program will be considered a success once we have evaluated that the program as developed matches the proposed project, and once that has been written into the final report.

# References

This is a list of the outside sources used in this paper.

Klemenc, Stacey Enesey. (2013). Cribbage: It’s not just a game, it’s an obsession. Retrieved from <https://www.militarynews.com/norfolk-navy-flagship/entertainment/on_liberty/cribbage-it-s-not-just-a-game-it-s-an/article_fb6049b2-7b72-50ad-88df-cbb6e9decad7.html>

The American Cribbage Congress. (2018). Rules of Cribbage. Retrieved from <http://www.cribbage.org/NewSite/rules/default.asp>

Lumetta, Steve S. (Accessed August 31, 2018). Amusing Cribbage Facts. Retrieved from <https://web.archive.org/web/20080516084713/http://www.crhc.uiuc.edu/~steve/humor/cribbage.html>

Cribbage Corner. (2010) Facts about cribbage. Retrieved from <http://cribbagecorner.com/facts>