Software Requirements Specification for 4ZP6: DieSpy

Team #9

Jackson Cassidy
Wyatt Habinski
Christian Majid
Paul Puscas

October 11, 2024

Revision History

Date	Version	Notes
October 11, 2024	0	Initial SRS documentation
February 3, 2025	1	Updated P3’s
April 4, 2025	1.1	Updated document to be in line with other documents and the final state of the project

Table of Contents

Personnel

Name	Email	Role
Jackson Cassidy	cassij2@mcmaster.ca	Project planner, Developer
Wyatt Habinski	habinskw@mcmaster.ca	Project planner, Developer
Christian Majid	majidc@mcmaster.ca	Project planner, Developer
Paul Puscas	puscasp@mcmaster.ca	Project planner, Developer

Naming Conventions and Terminology

Terms and Acronyms

- D6/D4/D8/D12/D20: A dice with that corresponding number of sides. For further information see https://en.wikipedia.org/wiki/Dice_notation
- ML - Machine Learning
- CNN - Convolutional Neural Network
- DND, D&D - Dungeons and Dragons
- WH, 40K - Warhammer

Definitions

- Tabletop games: There are 2 general types, tabletop war games, and tabletop roleplaying games. They are generally played in person, with all participants sat around a table (hence the name), and almost all require dice to be rolled to resolve in-game actions. The amount of dice depends on the game, but many of the most popular can require 10+ dice to be rolled by a player in their turn.

Stakeholders

Direct Stakeholders

- Warhammer Players
- DND Players
- Other tabletop gamers that require dice

Our direct stakeholders are tabletop gamers that use a multitude of dice. This is why gamers that play games such as warhammer and D&D are our main focus. These

games are known to use a very high number of dice, and during a fast paced game, they will be our main users and the direct stakeholders.

Indirect Stakeholders

Our indirect stakeholders are the TA's and professor.

User Participation

This app will be created in collaboration with the tabletop community. We will continuously gather feedback and implement quality of life changes to support the players as best as possible.

Constraints

Schedule Constraints

- Project Plan - October 25, 2024
- Proof of Concept - November 21, 2024
- Design Document - Jan 24, 2025
- Software Verification and Validation - February 7, 2025
- Final Demonstration and Video - March 25, 2025
- Final Product - April 4th, 2025

Application Constraints

App is only being released on the Google Play Store for now, as it is harder to enter into the Apple App Store. It costs more to register as an apple developer being $100 USD annually, whereas the Play Store is only a $25 USD one-time fee. The registration process on Google Play is also much simpler in regards to the Play Store. Apple in general has a stricter guideline with the app submission

Relevant Facts and Assumptions

Relevant Facts

- Tabletop games require large amounts of dice to be rolled each turn

Assumptions

- Users have dice
- Users have a table
- Users have an android device

Functional Requirements

P	Requirement	Details
P0	Camera functionality	The app must be able to obtain camera images of the dice used. The simplest way to do this will be through discrete camera inputs, using CameraX libraries for Android. The intended minimal camera use will consist of a top down photo of the board.
	Dataset	The application must have a fully functional dataset of dice for each model obtained. While this will not be a part of the application per se, it is nevertheless an extremely high priority task. See Data and Metrics for more details.
	Detect and Identify D6s, Count number of D6s, Sum D6	The application must be able to identify as many D6s as are on screen, detect what the value of each die rolled is, and aggregate the results by summing the number of each value rolled.
	Offline mode	By default, the camera and dice detection screen should run without any network connection required.
P1	Scan from Live Video Feed	The app should be able to take a continuous stream of video from the camera and determine which frames to process through the agents. Additionally, the app should not send repeat results after the dice have come to a stop (after a roll but before the dice are picked up).
	Login screen, option to Host or Join	The app should have the option to choose between hosting and joining. As well as a screen to connect securely.
	Connect to host and view results	The app should be able to allow hosts and guests to connect using a code, provided they are in physical proximity. Connected guests should have the dice roll results displayed in their app.

	Maintain account/turn system, i.e track rolls by who's turn it is	The app should be able to increment a "turn" through manual input (such as clicking a "Next Turn" button), and should assign the results of a series of rolls to one user.
P2	Log dice roll results	The app should have the ability to show users a history of previous rolls in the current session. This should be transient over games, and should carry over between sessions.
	Save previous game history	The app should keep track of previous players games, and be able to pick up where they left off, or simply go back to view game results
	Editing dice roll (in case of mistake)	The app should give the host the option to approve or deny or change a dice roll result before it is sent to the guests.
	Statistics by account	The app should be able to present aggregate scores to each user as to the rolls that were part of their turns.
	Live Chat	The app should have a built-in chat function, allowing users to send text messages to each other during the session. Live updates should also be sent to the chat.
	Simulate Roll option	The app should have a feature to simulate rolls without having to roll them physically, to save time.
P3	Highlighting dice in camera mode	The app should have a toggleable option to highlight dice in the image using the bounding boxes and results produced by the ML agents.
	D4 Support	The app should extend support to recognizing and identifying D4 Dice. This is a technical challenge, as D4s are the only dice with multiple numbers per face.
	Send live footage to guests too	The app should share the live video feed of the host's camera with the guests. The user will be able to toggle between this view and the traditional results view.

	Upload to google play store	The app will be published on the Google Play Store, not only to demonstrate its development quality and robustness, but also to enhance accessibility for users.
	D20, D12, D10, D8 Support	Added support for polyhedral dice beyond the D6. If possible, all supported dice types could be rolled and processed together. Otherwise, the option to switch between dice types will be present.

Data and Metrics

Dice Detection Agent

- Based on YOLOv10 this agent will take an image from the device camera and both draw a bounding box around each die, and correctly identify the side facing up.
- Will be trained on images of dice on a variety of surfaces. The number of dice in each image will range from 1 - 40+.
- Our goal is to have it identify both d6's with dots and ones with numbers. Should the product be extended to multiple dice types, the dataset will reflect this.

Dataset

- Link to current dataset here
- An imageset has been started, and will continue to grow. We will be aggregating a series of existing datasets, as well as extending it by rolling dice and taking pictures (with sufficient variety of surface, dice, angle, lighting, etc.)
- We anticipate approximately 100 images of multiple dice as an absolute minimum for a functional demonstration.
- The dice in these images must be labeled with bounding boxes as well as class labels.
- Data augmentation will also be used to apply transformations such as rotation, scaling, flipping, and colour adjusts to expand our dataset.

Metrics

In terms of classification metrics, we will aim for >95% overall accuracy, and a peak F1-Score of >0.9. We chose these as metrics since F1-Score balances precision and recall, precision to measure how often the model is correct when it predicts a specific class, and recall to measure how often the model detects and classifies a

die correctly. We chose accuracy because we have balanced classes, false positives and false negatives are of equal importance in this task, and overall “correctness” is a good representation of the performance goal of our model.

For bounding boxes we will aim for mAP50 (mean average precision with an intersection over union threshold of 0.5) >0.9, and mAP5-95 (mean average precision with intersection over union thresholds ranging from 0.5-0.95) >0.75. We chose these because very precise object location is not a priority, and these balance class detection with bounding box accuracy, which suits our goals better.

Non-Functional Requirements

Look and Feel Requirements

Appearance Requirements

- Simple tabulated UI that has a clear and intuitive flow which guides the users without needing prompting.
- Fluid camera.

Style Requirements

Maintain a consistent style throughout all pages of the app. This includes fonts, colour scheme, and layout.

Usability and Humanity Requirements

Ease of Use Requirements

The application should be self-explanatory. Users should be able to either 1) intuitively understand, or 2) trivially discover the functionality of each available feature.

Learnability Requirements

- Users will be required to learn the UI.
- If users wish to join a host, they will have to understand where to find the Party’s code and how to input it.

Understandability, Politeness and Age Requirements

The app will have no vulgar, sexual, or explicit language. This app will be suited for all ages.

Accessibility Requirements

- The application UI will use an easily intelligible font and colours. Colour blindness should be considered when deciding on a colour scheme.

- The results of dice rolls should be available on screen as text.
- Will be available on all sizes of android screens. (tablets, and phones)

Performance Requirements

- Dice recognition should be faster than summing/counting the dice yourself.
- Our application must not cause crashes or phone damage due to overheating.
- The application should not crash during nominal operation. Any network issues should be handled gracefully.
- The application should be able to identify and correctly classify over 30 D6 dice. If other die types are included, it should be able to classify those as well.
- The application should support up to 10 simultaneous, lag free connections for party play.
- Barring updates to the Google Play Store, and the APIs used, our app should continue to function indefinitely.

Precision and Accuracy

- >95% accuracy
- Peak F1-Score >0.9
- >0.9 mAP50
- >0.75 mAP50-95

Cost Requirements

- Cost of dice for testing
- Cost to publish to google play store
- Potential need for purchasing server space

Security and Privacy Requirements

Access Requirements

We require access to cameras, device-to-device network connections, servers for login features. Secure wifi connection, and login features will not allow for unauthorised third party access

Integrity Requirements

The app must be compliant with Google Play Store's Code of Conduct

Privacy Requirements

No identifiable data should be collected.

Audit Requirements

The application should be auditable to comply with Google Play Store requirements.

Immunity Requirements
We require a licence that absolves the developers. It optionally should be compliant with Google Play Store Regulations.

Operational and Environmental Requirements

Expected Physical Environment
It is expected that the app will be taking pictures of a flat table or surface, with multiple dice on top. The angle of the camera will primarily be from above (vertically). If possible lower camera angles should be attempted.

Requirements for Interfacing with Adjacent Systems
The application should ideally be able to interface with other Android devices using the application.

Legal and Compliance Requirements

The app will use a licence that legally insulates the developers. If brought onto the Google Play Store, the licence must be with that storefront.

The app should, but is not obligated to, follow ISO standard wherever possible.

Maintainability and Support Requirements

Maintenance Requirements
The Application should not require maintenance, barring updates to the APIs being used.

Supportability Requirements
The Application must be developed for Android and will be available for use on Android devices.

Adaptability Requirements
The application will be modularized allowing for seamless updates.

Off-the-Shelf Solutions

Ready-Made Products

- D&D Beyond
- Owlbear rodeo
- Shard Tabletop

- Roll 20
- Fantasy Grounds
- Tabletop Simulator

Can these Products Be Copied?
None of the products listed as ready-made products do what ours will do. These are environments for playing virtually and/or tracking system specific content, but most of the fun comes from rolling physical dice. This app aims to encapsulate that, without getting bogged down by having to add dozens of six-sided dice.

Effects on the Install Systems

- Requires access to Camera on personal device
- May need access Bluetooth to join sessions locally
- May need access to the internet to join or create any party

Project Planning

Airtable will be used for project planning and task allocation, both for short term and long term tasks. We will continuously take meeting minutes after all meetings to ensure we are all aware of the scheduling and pace. Airtable will be used to track sprints and their tasks

Risks and Issues

Predicted

Data Collection:

- Diverse Environments: Collecting data in controlled environments might not capture real-world variability (e.g., lighting conditions, dice occlusion). Inability to capture diverse datasets may reduce model robustness.
- Incorrect Labelling: Errors during the data labelling process could lead to incorrect predictions by the model. Mislabeling dice faces or bounding boxes may reduce model accuracy.

Technical Issues:

- Model Performance: The model may fail to accurately detect dice or misinterpret numbers, especially with non-standard dice shapes/colours.

- Data Scarcity: Insufficient or biassed data may lead to poor generalisation of the model, reducing its effectiveness.
- Accuracy in Crowded Images: Since the app must handle a large quantity dice, predicting the correct dice number in cluttered scenarios may be challenging, leading to incorrect classifications.

Operational:

- Real-Time Processing: Implementing real-time dice detection and number identification may result in latency, leading to a poor user experience.
- User Hardware Variability: The app's performance may vary across different devices, especially those with lower processing power or inferior cameras.

App Rejection by Google Play Store

- Google Play Policies: Failure to comply with Google Play's policies (e.g., privacy, security, data usage, content) could result in the rejection of our app.

Open Issues

None as of April 4th, 2024