

Project Requirements Document v1

Project Title: RealtyReel

Team Name: Brandon and The Verilog Survivors

Company: AppFolio

Authors: Jacob Eisner (Team Lead), Iain Weissburg (Scribe), Liam Pang-Naylor, James Ellenstein, Yuki Keung, Brandon Son

Introduction

Background

Property managers use various strategies to advertise their properties effectively. Advertising helps attract potential tenants and ensure the property's occupancy rates are consistent. Traditional in-person property visits often come with challenges, including scheduling conflicts and traveling inconveniences. In contrast, video tours offer several advantages that make them a preferable alternative. Video tours offer a more interactive and detailed experience compared to static images and textual descriptions typically featured on rental websites. These videos provide a comprehensive overview of the property, showcasing key features in a condensed format.

In order to capture younger audiences, short form video contents may become the preferred method of advertising. Short form video contents capture viewers' attention and interest quickly, while offering a snapshot of the property's key features and highlights. Property managers can thereby cater to the preferences of today's audience, who favor easily digestible and visually captivating content. However, the process of creating high-quality short form video tours, while effective, can be time-consuming. RealtyReel addresses this issue by automating the entire video production process.

Problem Being Solved

RealtyReel aims to solve the time-consuming and often expensive process of creating professional-quality property listing videos. Traditional methods involve hiring videographers, editors, and voice-over artists, which can be prohibitive for smaller property owners. RealtyReel's innovation lies in the AI algorithms that autonomously edit and assemble the media, thus offering a streamlined and cost-effective solution. The core technical advances include machine learning algorithms for video editing, natural language processing for generating voiceovers, and AI-based color grading techniques.

High-quality visual media can significantly enhance the appeal of property listings, making it easier to attract potential buyers or tenants. However, the investment required to produce such media can be a significant barrier for many property owners. By automating this process, RealtyReel democratizes access to high-quality property marketing, potentially accelerating sales and rentals, which has a broader positive economic impact.

Currently, the problem is largely solved by hiring professionals in videography and editing, or by property owners taking a DIY approach that often results in less polished content. There are also some software tools available for video editing, but they typically require a degree of skill and time investment that many property owners lack. RealtyReel offers a third option that combines the quality of professional services with the convenience and affordability of a DIY solution.

Assumptions

For our project, RealtyReel, we make several key assumptions. First, we assume that property managers have access to images, videos, and descriptions of their properties. We also assume that the users will provide high-quality images or videos for best results. Lastly, we assume that the AI algorithms used for script generation, voiceovers, and other features will produce accurate, non-offensive results.

Goals & Objectives

By the end of this project, we intend to have a working application that consists of an intuitive UI alongside a functional backend that is able to produce high quality videos for our users.

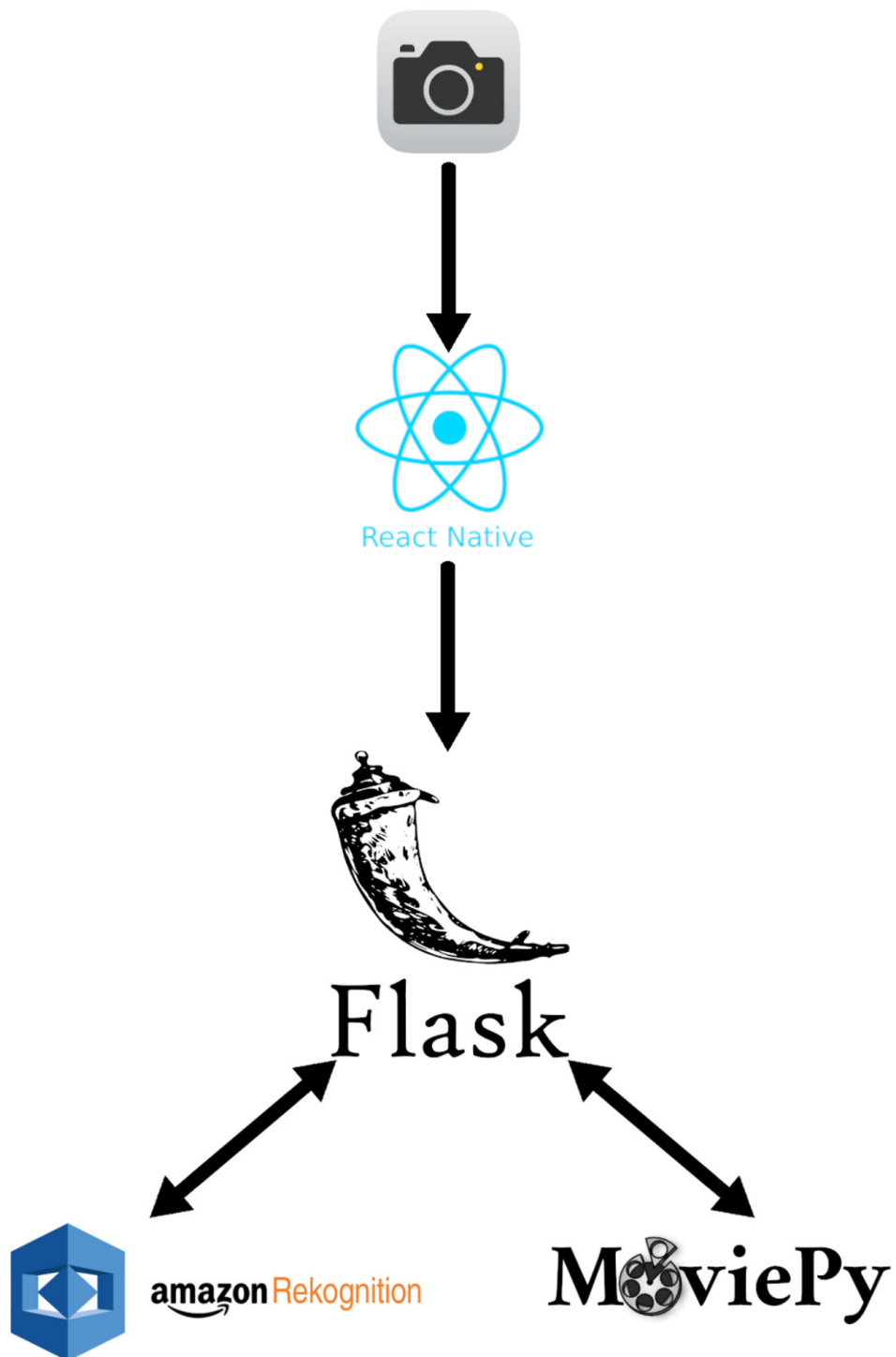
Minimum Viable Product	<ul style="list-style-type: none"> - Frontend App - User uploads video - User fills out a form: tag selection, general property info - Basic video edit and rendering - Video should be ~30 seconds
Feature Set 1	<ul style="list-style-type: none"> - Script generation - User accounts - User text overlay
Feature Set 2	<ul style="list-style-type: none"> - AI Voice over - User friendly configuration options/sliders
Feature Set 3	<ul style="list-style-type: none"> - TikTok/YouTube Shorts integration

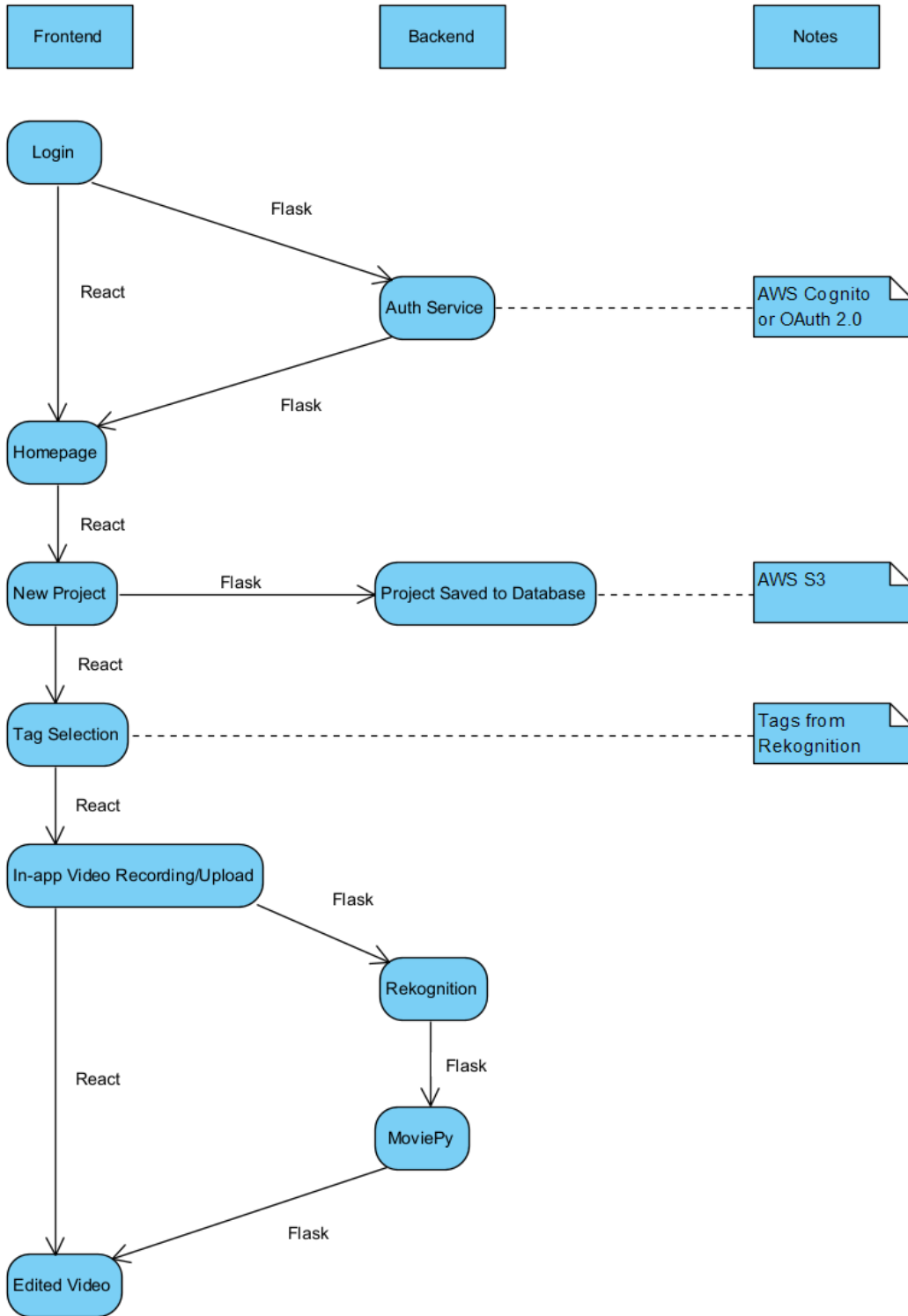
Project Specifics

For this project, we intend to use React Native for our frontend and Flask for the backend with a PostgreSQL database. The user will upload photos and videos of the property which will be interpreted by the open source multi modal LLM LLaVA (future feature). LLaVA will caption the images and Google text-to-speech will create a voiceover. Based on the length of the advertisement we will use AWS Rekognition to edit, color correct, and render the video based on a predefined template. Other offerings such as AWS Cognito and AWS EC2 may be used for OAuth2 login and Backend hosting respectively (future feature).

System Architecture Overview

High Level Diagram





User Interaction and Design

1. User opens web app.
2. User can login/create an account. (Optional).
3. User selects “Create New Project” button.
4. User provides title, address, etc for rental.
5. User selects relevant tags for their rental.
6. User uploads or takes a video walkthrough of their rental.
7. Web app provides edited video with download and share links.

Requirements

User Stories/Use Cases

As a new user, I can create a new account.

- Scenario 1: The user clicks the “Sign Up” button, enters valid registration information, and successfully creates an account.
- Testing: We can verify account database entries are correct.
- GitHub Issue:

As a user, I can log in and log out of my account.

- Scenario 1: The user clicks the “Login” button, enters valid login credentials, and successfully logs in to their account.
- Scenario 2: The user enters incorrect credentials and receives an error message indicating that the login credentials are invalid.
- Scenario 3: The user clicks on the “Sign Out” button and is logged out of the account.
- GitHub Issue:

As a user, I can create a new project.

- Scenario 1: In the home page, the user can click on the “New Project” button to create a new project.
- Testing: Verify project is created with default parameters.
- GitHub Issue:

As a user, I can see all my project videos displayed on the home page for quick and convenient access.

- Scenario 1: In the home page, the user can view all their previous project videos and click on them to watch them.
- Scenario 2: In the home page, the user has not created any project videos so there will be no videos shown.
- Testing: Verify projects/videos within database are equivalent to those displayed.
- GitHub Issue:

As a user, I can select tags for my project.

- Scenario 1: On the tags page, there will be a variety of options from AWS Rekognition to assist the AI in prioritization (i.e. Living Room, Furnished, Bedroom, etc).
- Testing: Verify selected tags are sent through backend to AWS Rekognition.
- GitHub Issue:

As a user, I can upload videos, images, and text descriptions. Alternatively, I can record videos in-app.

- Scenario 1: The user will see an upload or record option with the ability to upload a file (max size pending) or record a walkthrough using an embedded link to their native camera app.
- Testing: Test that both a file upload and embedded camera record a video and successfully send it to the backend.
- GitHub Issue:

As a user, I can easily navigate between pages when completing the project form.

- Scenario 1: On each page of the form, there are “Back” and “Next” buttons where the user can click to navigate back and forth between pages.
- GitHub Issue:

As a user, I can save my progress in the project form and pick up where I left off at a later time.

- Scenario 1: On each page of the form, there is a “Save for Later” button that the user can click on to save their progress.
- Testing: Ensure proper state is saved within database under proper account information.
- GitHub Issue:

As a user, I can view and save the final video.

- Scenario 1: Once the video processes, the user will be provided with a playable preview and confirm and retry options.
- GitHub Issue:

As a user, I can export my project videos directly to TikTok or Youtube or download the file.

- Scenario 1: After confirmation, the user will be presented with a download link along with TikTok and Youtube Shorts export embeds.
- GitHub Issue:

Non-functional Requirements

- Ease-of-use
- Account not required
- Video can be previewed
- Video production should be completed in a timely manner
- Data integrity and privacy for users

Appendices

Technologies Employed

- Frontend: React Native
- Backend: Flask
- Database: SQL
- Deployment platform: AWS
- User authentication: AWS Cognito
- UI design: Figma