

# **1. Introduction**

## **1.1 Purpose**

The purpose of this document is to provide a comprehensive overview of the requirements for the development of a startup funding web application.

## **1.2 Scope**

The web application will allow entrepreneurs to create profiles for their startups, submit funding requests, and connect with potential investors. Investors will be able to browse startup profiles, review funding requests, and make investments.

## **1.3 References**

<https://episyche.com/blog/how-to-integrate-razorpay-payment-gateway-in-django-and-react-for-the-checkout-use-case>

– Razor-pay Integration

<https://blog.learncodeonline.in/how-to-integrate-razorpay-payment-gateway-with-django-rest-framework-and-reactjs>

– Razor-pay Integration

# **2. Overall Description**

## **2.1 Product Functions**

The web application will consist of a backend API built with Django REST Framework (DRF), a frontend interface developed with React.js, and a MySQL database to store user and startup data.

## **2.2 Product Functions**

- User Registration and Authentication
- Startup profile creation and management
- Funding request submission
- Investment management
- Browse Startup and funding request

## **2.3 User Classes and Characteristics**

- Entrepreneurs: Users seeking funding for their startups.
- Administrators: Users responsible for managing the platform.

## **2.4 Operating Environment**

The web application will be accessible via modern web browsers on desktop and mobile devices.

## **2.5 Design and Implementation Constraints**

- Back end must be built using Django and DRF (Django Rest Framework)
- Front end must be built using React Js.
- Front end styling with Bootstrap.
- Integration with third-party gateway for handling investment like RazorPay.
- Integration with third-party library Django-cors-headers to connect with React Js and DRF.
- In Front end using axios library to access the endpoints of the DRF.
- Using react-hook-form library for managing complex forms.

## 2.6 Assumptions and Dependencies

- Assumption: Users will have basic internet connectivity and modern web browsers.
- Dependency: Integration with third-party APIs for email notifications and payment processing.

## 3. System Features

### 3.1 User Registration and Authentication

### 3.2 Startup Profile Management

### 3.3 Funding Request Submission

### 3.4 Investor Dashboard

### 3.5 Admin Panel

## 4. Database Design

### 4.1 Entity-Relationship-Diagram (ERD)

