

**CEN 4090L: Software Engineering Lab**  
**Florida State University**  
**- Group Project Proposal Template -**

**1. Project title**

Social Media Application: ListPix

**2. Brief overview of what you are proposing**

Social media application where the user gets a list of goals to complete (or a bucket list) of random activities when making an account. Users post pictures of themselves completing these tasks, aiming to complete as many bucket list items as possible. The platform fosters a sense of accomplishment and community by allowing users to track their progress, interact with others, and connect with users sharing similar goals. When you view a user's profile you can see all of the bucket list activities that they have completed and those that they have yet to complete.

**3. Motivation**

We want to create an application that encourages users to step out of their comfort zones and try new activities. We want to expand our knowledge of integrating Python with the Django framework and storing the data in Firebase. We are also highly interested in Social Media applications. Our goal is to learn about how to make an engaging and interactive platform.

**4. Features to be implemented and types of users**

User Registration: Roles (admin, contributor, and observer), Profile Setup

Randomized Bucket List: Initial list, Limited reroll ability for a fresh challenge

Content Posting: Task Completion, Interactive feed

Progress Tracking: Profile Display, Activity Dashboard

Search and Explore: User Profiles (see others progress), Activity Search (explore others who have similar goals)

Forum: Community to discuss tasks, tips, inspiration

Admin Features: Content moderation, DB Management

**5. Risk / Challenges**

Our challenge will be using languages and interfaces we don't have much experience with. We have only built local web applications, so a risk might be underestimating the workload involved.

**6. Existing related projects**

The Bucket App- Social media app which users create and share bucket lists. Our application is different because our application creates the bucket list idea for you. This allows users to explore options that they had never considered before. (link: <https://thebucketapp.com/features>)

Instagram- Social media app where users share pictures and experiences. Our application is different because it has a strict focus on exposing users to new experiences, whereas Instagram has a broader purpose. (link: <https://www.instagram.com/>)

**7. Intended platform / programming language**

Backend: Python (Django) - User data, task management, content posting, user interactions.

Frontend: HTML/CSS - Structure and Styling (possibly JavaScript for interactive elements - feed updates and reroll functionality)

**8. Third-party libraries / APIs to be used**

Firebase for data storage (user profiles, posts, bucket list items, forum discussions).

Django framework (authentication).  
Randomization library.

Deployment: Git for version control/collaboration and a cloud service for server

## 9. Team members, expertise, project responsibilities, and team organization

Olivia Judah:

- ogj21
- C++, python, js w/ angular, postgresql, html/css, front and backend integration for web app

Nicolas Caballero:

- nc19m
- C++, C, C#, SQL, XAML

Felipe Lima:

- fml21
- C++, some C, and currently learning python

## Collaboration

- Group message and weekly updates on task completion/issues
- Version Control: Merge into dev branch regularly once pull request and code review approved and tested by 2 people
- Task Management: Excel sheet to track tasks, bugs, and features.
- Document individual code as you go

## Timeline

### Project Milestones and Check-ins

- End of Week 2: Ensure the initial project setup is complete, and the database schema is designed.
- End of Week 4: The backend should be mostly implemented, with basic models, views, and CRUD operations in place.
- End of Week 6: The frontend should be functional with the core UI components and basic interactions working.
- End of Week 8: Core features should be complete, and the platform should be ready for user testing.
- End of Week 10: The application should be fully tested, polished, and ready for deployment.

## Week 1: Planning and Setup

### 1.1 Define Requirements

- Finalize the platform's core features and scope.
- Define project objectives

### 1.2 Research and Learning

- Identify resources for Django, HTML/CSS, and necessary libraries.
- Assign specific research topics to team members (e.g., Django user management, frontend frameworks).

### 1.3 Initial Project Setup

- Set up version control (e.g., GitHub repository).
- Set up development environments.
- Initialize a Django project and establish a basic structure.

### 1.4 Create a Project Plan

- Break down the project into manageable tasks and assign them to team members.
- Create a Gantt chart or similar project timeline tool to visualize milestones.

## **Week 2: Design Phase**

### 2.1 Mockups

- Design visual models for the main pages (user profiles, feed, forum)
- Determine aesthetic (colors, fonts, etc)

### 2.2 Database Schema Design

- Define data models for users profiles, bucket lists, posts, and forum.
- Create an initial database schema.

### 2.3 User Flow and Functionality

- Define user journey from registration to content posting and goal tracking.

## **Week 3: Backend Development - Initial Setup**

### 3.1 Django Models

- Implement Django models, views, and serializers for user authentication, bucket lists, forums, and posts
- Set up migrations and create the initial database.

### 3.2 User Authentication

- Implement user registration, authentication, and profile setup

### 3.3 Task Randomization

- Develop logic for generating randomized bucket list tasks upon registration and rerolling goals.

## **Week 4: Backend Development - Core Features**

#### 4.1 Content Posting and Feed

- Implement content posting features and interactive feed.
- Integrate like, comment, and share functionalities.

#### 4.2 Progress Tracking

- Develop logic to display completed and pending tasks on user profiles.

#### 4.3 Testing and Debugging

- Perform initial testing and debugging of core backend features.

### **Week 5: Frontend Development - Basic Setup**

#### 5.1 HTML/CSS Layout

- Develop the basic structure and styling of the web application.
- Create templates for core pages (home, profile, forum).

#### 5.2 Navigation and User Flow

- Implement basic navigation between pages.
- Ensure the user flow is intuitive.

#### 5.3 JavaScript Integration (If needed)

- Implement interactive elements like feed updates, posting, and reroll functionality

### **Week 6-7: Frontend Development - Core Features**

#### 6.1 User Profile Display:

- Implement the profile page showing task progress and completed tasks.
- Develop search functionality to explore other users' profiles.

#### 6.2 Forum Development:

- Build a forum where users can connect and discuss goals.
- Integrate with backend for posting and interactions.

#### 6.3 Testing and Refinement:

- Conduct thorough testing of frontend and backend integration.

### **Week 8: Extra Touches**

#### 8.1 Finalize Progress Tracking:

- Complete the progress tracking features.

#### 8.2 Forum Enhancements:

- Add features like pinned posts, categories, and user mentions.

#### 8.3 Testing and Debugging:

- Conduct extensive testing of all platform features.

### **Week 9: User Testing and Feedback**

#### 9.1 User Testing:

- Conduct user testing sessions to gather feedback on usability and functionality.

#### 9.2 Bug Fixes and Improvements:

- Address any issues reported during testing.
- Finalize features.

#### 9.3 Documentation:

- Begin creating user and technical documentation.

### **Week 10: Deployment and Launch**

#### 10.1 Deploy Application

- Deploy the platform to the production server.
- Perform final checks and resolve any issues.

#### 10.2 Launch Event and User Onboarding (OPTIONAL)

- Plan a soft launch or beta testing phase with select users?

#### 10.3 Finalize Documentation

- Wrap up documentation and report

### **Week 11: Final Presentation and Wrap-Up**

#### 11.1 Project Wrap-Up

- Finalize and submit all project documentation.
- Prepare a final report detailing the development process, features, and outcomes.

## 11.2 Final Presentation

- Present the project, highlighting key features and the focus on user engagement through goal completion.