

# CozyFolio

Dev Soor, Felix Almejo

## Section 1: Summary

Application is called **CozyFolio**

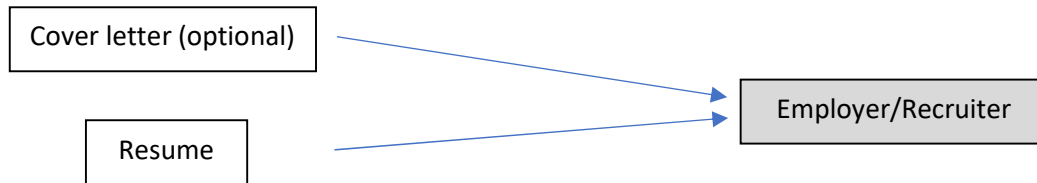
The target audience is software engineers, developers, product managers and technical sales candidates looking for a job. Currently, a candidate looking for a job must submit various types of information, such as portfolio of projects developed, LinkedIn profile link, resume/CV, GitHub account link, Stack Overflow etc. The candidate may have a dedicated website which showcases their projects and maybe even include a resume, but all other information is scattered and not easily available or visible to prospective employers and recruiters. Our platform will allow the users to consolidate all this information in one place where they can upload documents, create multiple portfolios, upload project information such as Youtube videos, screenshots, descriptions etc. The final deliverable to the candidate/user is a link that they can send to a prospective employer/recruiter that contains the consolidated information. The biggest benefit to the user is that it saves them time, and ensures no information is left out and is cost effective.

The primary audience for this project is software and web developers, which we call “users”

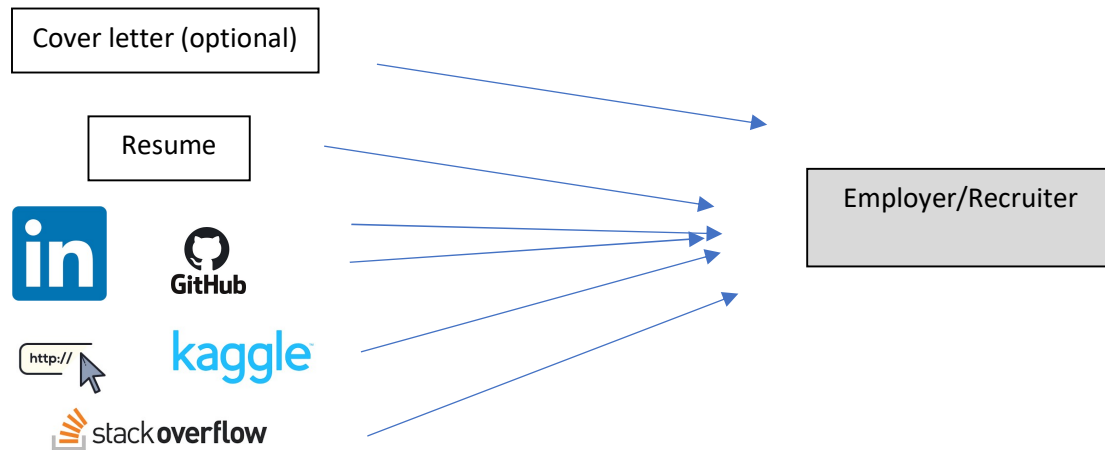
# CozyFolio

Things you need to get hired as software developer

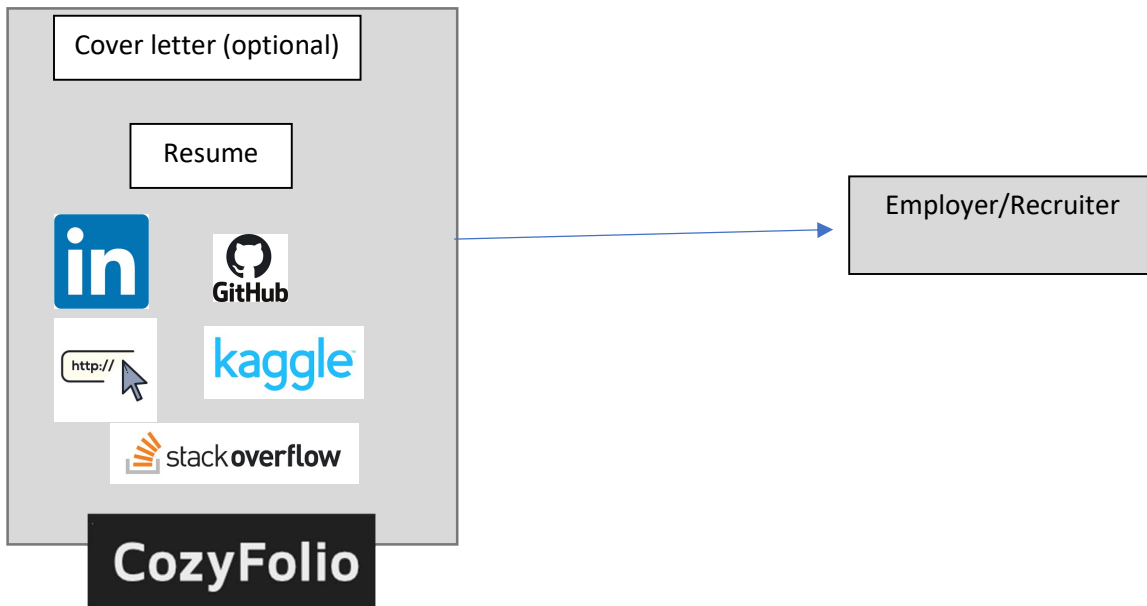
A long long time ago .....



TODAY .....



FUTURE .....



# CozyFolio

## Section 3: Feature List

- User will be able to create an account and log in using registration information
- User will be able to see their profile on login
- User will be able to create and update their profile, with personal information, skill set, domain name etc...
- User will be able to add/remove Portfolios (geared toward job prospect).
  - User can add and update information for the portfolio
- User will be able to add/remove Projects that are linked to desired portfolios
- User will be able to link github, linked-in, stack-overflow, kaggle, portfolio website

## Priorities:

### P0:

- Create Login/ Registration for user to gain access
- Dashboard page for users to manage their portfolios and projects.
- Profile page for specific information about the user, skills, tech etc.
- Portfolio page to add/delete projects and upload resumes
- Projects Page for user to add/ remove projects to be linked to a specific portfolio
- create ONE website (portfolio page)
- Define Models for user portfolio and projects (models.py)
- define URL paths (urls.py)
- write functions in views.py file

### P1:

- Admin user page to be able to manage users (table)
- have multiple website templates
- make skills global to the website

### P2:

- Create employer/recruiter login/registration
- Create employer/recruiter dashboard
- Allow recruiters to search for candidates (users, skills, languages, etc)
- Show stats about users

## CozyFolio

- Most popular languages used by users
- Most popular sought-after job title (full-stack, front-end, QA, PM)

P3:

- Google maps showing user location to employers
  - showing language/ frameworks / DB hot spots
  - job-title hot-spots
- Assist candidates with interview process (track, deciding, facilitate communication)
- Resume Generator
- LinkedIn Profile Generator

## Section 4: Technical Considerations

- Backend
  - Python
  - Django, sql-lite3 (postgresql,if time allows)
  - bcrypt
  - Pillow
- Front-End
  - HTML/CSS
  - Bootstrap4
  - Material Dashboard
- Deployment
  - amazonAWS
  - NGINX
  - Gunicorn
- Other
  - Google Maps API
  - LinkedIn API
  - GitHub API
  - Plotly, D3 Charts

# CozyFolio

## Section 5: Work Distribution

P0:

- Create Login/ Registration for user to gain access - Dev
- Dashboard page for users to manage their portfolios and projects. - Dev
- Profile page for specific information about the user, skills, tech etc..- Dev
- Portfolio page to add/delete projects and upload resumes - Dev
- Projects Page for user to add/ remove projects to be linked to a specific portfolio - Dev
- create ONE website (portfolio page) -Dev
- Define Models for user portfolio and projects (models.py) -Felix
- define URL paths (urls.py) - Felix
- write functions in views.py file -Felix

P1:

- Admin user page to be able to manage users (table) -Dev/Felix
- have multiple website templates – Dev
- make skills global to the website - Felix