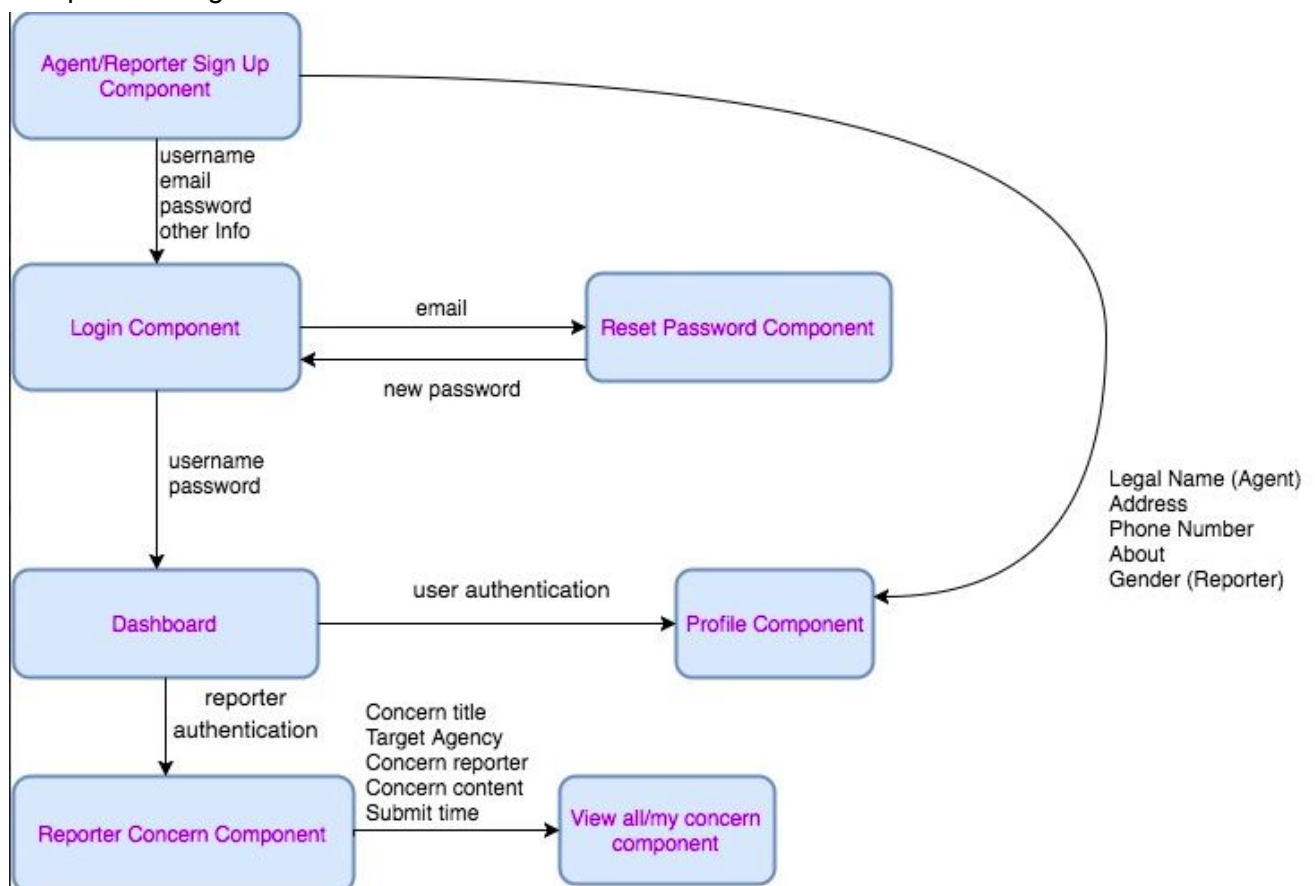


1.1 Define all components - 20 pts

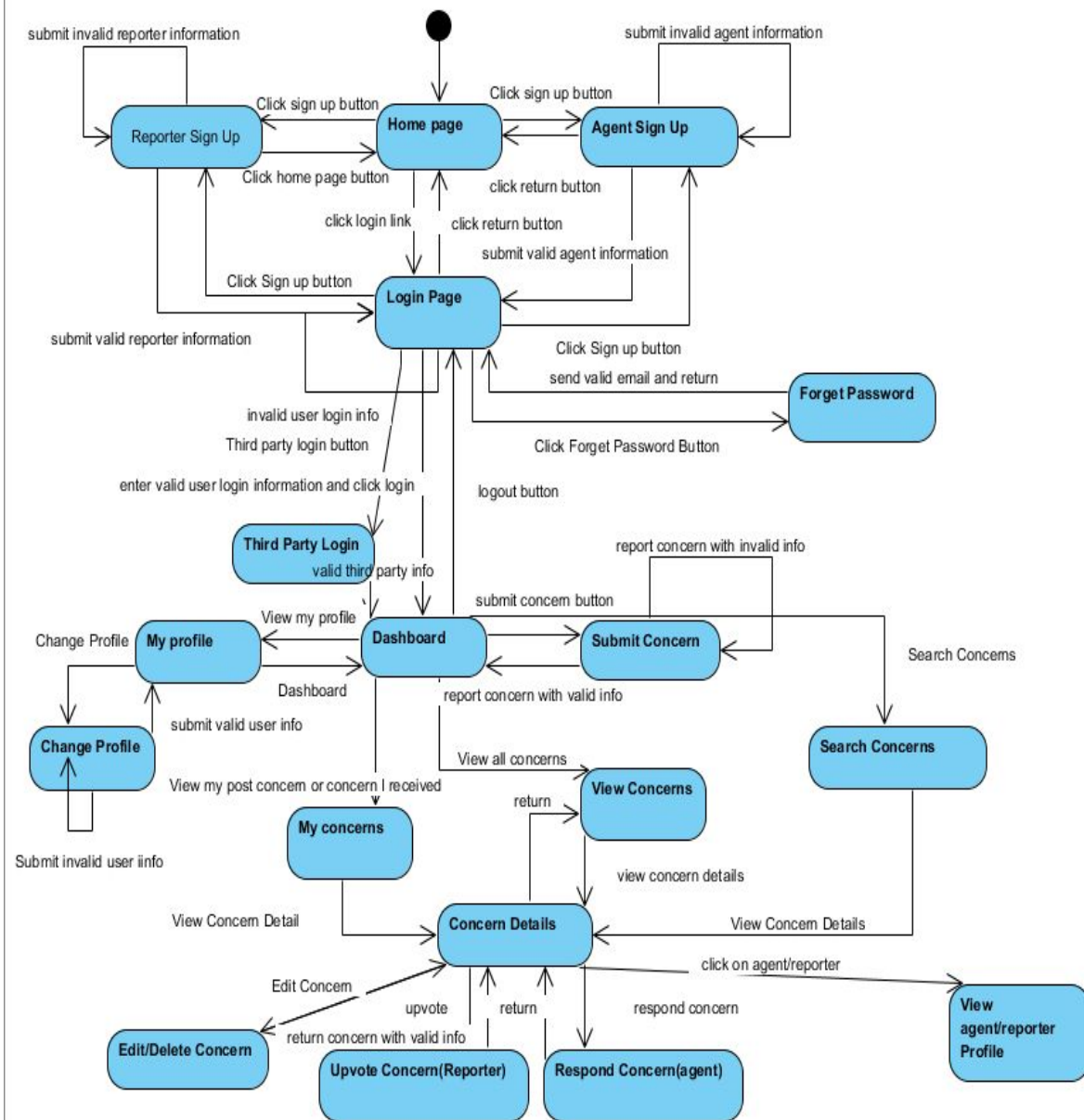
- We expect a **diagram** explaining various components in the project. You should outline the base components and other independent components, and show the interaction among the components - 10 points
- Specify the **input, output and the dependent components** (components it calls and the components that call it) for each component - 10 points

1. Login component
2. Agent/Reporter Signup component
3. Dashboard component
4. View my/all concern component
5. Profile component
6. Report concern component
7. Reset password component

Component Diagram



Flow Diagram



Components:

1. Login(depend on agent/reporter sign up):

- password (required, match)
- username (required, match)

2.1 Reporter Sign up Component:

- username(unique, required)
- email(unique, required, match email format)
- password(required, password should be strong)
- legal name(optional)
- address(optional)
- phone number(optional)
- about(optional)

2.2 Agent Sign up Component:

- username(unique, required)
- email(unique, required)
- password(required)
- legal name(required)
- address(required)
- phone number(required)
- about(required)
- Agent Verification File(required)
- Agent Badge Logo(required)

3. Dashboard Component(dependent component on login):

- login (required)

4. View all/my concern Component(depend component on Report Concern Component)

- title(required, maxlength)
- timestamp(required)
- reporter(required, depend on reporter sign up)
- agents(required, depend on agent sign up)
- status(required)

5. Profile Page(all dependent component on agent and reporter sign up):

- legal name(required for agent)
- address(required for agent)
- phone number(required for agent)
- about(required for agent)

6. Report Concern Component(dependent component on agent and reporter sign up):

- title (required, maxlength)
- target agent (required, depend on sign up agent)
- content (required)
- reporter(depend on login user)
- timestamp(required)
- upvote number(required)
- response(optional)
- status(required)

7. Reset Password Component(depend on user/reporter sign up)

- Email(required, depend on sign up)

1.2 Form of incremental testing - 5 pts

We decided to use **bottom-up incremental testing**.

Because:

- It would be more straightforward to create testing input concerning about certain buggy function instead of a whole module.
- It would be a lot easier to observe testing output to confirm the elimination of certain bugs.

2 Incremental and Regression Testing (70 points)

- Incremental and regression testing should be conducted on a **per module basis**. We will be looking for testing to be completed on each of your modules. We expect you to list down the defects found during incremental testing and regression testing of **each module**. The defects will be evaluated based on how well the defect is described, the severity assignment and the solutions provided.

Increment Test

Defect No.	Description	Severity	How to Correct
1.	The web page does not complain when "legal_name" is not entered	1	Modify the models.py to make "legal_name" field required
2.	The submitted time for the concern submitted should be properly recorded	1	Modify the models.py to add "upload_time" field

3.	The website does not regulate on password's level of complexity, and makes user's password potentially endanger.	2	Password strength check is made more strict
4.	When a duplicated email is used for signup/reset password, the signup/reset password can be approved.	1	Signup/reset password will be rejected if duplicate email address is used
5.	The website is unable to properly reflect the targeted agent when viewing "my submitted concern" or "all concern" (when excess amount of targeted agent is presented).	3	Modify the frontend page and find a better way of representation for large set of targeted agents. (Dropdown list, for example)
6.	When signing up as an Agent, the verification file/logo file link are not checked to see if the file is in correct format	2	Do a string check on the urls provided, to see if it have a valid file extension
7.	The concern submission page does not have upload file feature supporting concerns	3	Add a file upload feature in concern submission page
8.	The user is able to enter invalid agent name in the submission of concern.	3	Implement a drop down list with search functionality to replace the current text input.
9.	Profile page does not correctly reflect user's information.	3	Correctly implement the profile page that supports and displays Django template
10.	The user is allowed to access the dashboard without logging in	3	Redirect the user to 404 page when not logged in
11.	Click on "view all concern" tab will lead to program crash	3	Display all concerns submitted by all registered users in all concern page

Regression Test

Defect No.	Description	Severity	How to Correct
------------	-------------	----------	----------------

1.	Making the “legal_name” field required causes the profile page to crash on loading	1	Provide a default value for historical entries in the database, and require user to add the value in profile page
2.	The new concern submitted has a default value of “False” on targeted agent, but it shouldn’t be	1	Modify the models.py to make “legal_name” field not having default value
3.	There exists a possibility of malicious user guess the right password even after the enforcement.	1	Set up a rate limit for each request input into the server. Stop accepting request from certain IP address if it is determined that certain address has sent way too many requests.
4.	Fixing duplicated email cannot fix history records with duplicate emails are still in database	3	Create a bad email list, and when this user login, prompt him for reset email
5.	If duplicated email is fixed, when clicking “forget password” and sending email to the original duplicated email, the email sent will fail because of duplicated same email.	3	Create a bad email list, and when this user login, prompt him for reset email to fix this bug.
6.	Adding the file upload feature leads to possible upload defects	2	Be careful of upload feature implementation, like checking the file format, file size
7.	Profile page crashes when an admin account try to enter profile page	1	Add a check to see if current user is admin and show corresponding content
8.	When not logged in user is redirected to 404 page, the user may be confused about where to go	2	The user is redirected to login page when he or she has not logged in and try to access dashboard
9	All concern page does not contain timestamps, votes etc. features	3	Add those statistics to all concern page.

3 Update Product Backlog

- Update the status of the requirements in your product backlog document:
- If a requirement planned in the **current sprint is complete**, change its status to ‘**Completed in sprint**’

- If a requirement planned in the **current sprint is incomplete**, change its status to **'Incomplete: moved to sprint '**
- If a requirement planned in the current sprint is partially complete, change its status to **'In-progress: moved to sprint '**
- If a (not yet planned) requirement is selected for the next sprint, mark its status as **'Planned for sprint**

