SocialWorker Release 2 Summary

Team members

Name (Student id)	GitHub id	Story points
Adrianna Diaz (27184778)	**adriannadiaz**	**6**
Liuai Hatter (25976618)	ldhatter	6
Anna Rogozin (27494939)	annarog	7
Francis Bouchard (26786812)	francisbouchard	5
James Talarico (40008054)	jtalarico	3
Wahab Wajahat (21311980)	wahabwajahat	3

Project summary

A tool to facilitate the management of notes that frontline social workers take when meeting participants (the people who they help). The software runs as a secure web application and has access to an internal encrypted database. The web application is able to sync and fetch relevant case files for the participants. To assist the frontline workers in the field, the application can adapt to any device.

Velocity

Iteration 6 (1 story, 13 points)
Iteration 7 (5 stories, 15 points)
Iteration 8 / Release 2 (4 stories, 18 points)

Velocity after Iteration 8 (Release 2): (13+15+18)/3 = **16**
We completed four user stories for a total of 18 SP during this iteration, which is above our previous average velocity.

URL: https://github.com/adriannadiaz/SocialWorker

Previous Project: DeepEye

We previously had a change of stakeholders, so the following is a summary of the implemented work for DeepEye.

Project Total: 18 stories, 101 points over 13 weeks

Iteration 1 (1 story, 8 points)
Set up frameworks and environment. Created a desktop application prototype to set directory of images to be used in the image classifier.

Iteration 2 (3 stories, 23 points)
Implemented the user interface so user can adjust parameters and labels of images to classify.

Iteration 3, (3 stories, 26 points)
Implemented an image classifier prototype using TensorFlow.

Release 1 Total: 7 stories, 57 points over 10 weeks
Release 1 aka Iteration 4, (4 stories, 23 points)
Integrated the image classifier prototype with the user interface. Application allows user to select images to train on, create and store models, and display classification prediction on an image.

Iteration 5, (21 stories, 21 points)
Application now returns predictions on multiple images. It was during this iteration that we left the stakeholders, Teledyne.

Current Project: SocialWorker

After changing stakeholders, the following describes work completed for SocialWorker. Note: once changing projects, we used user stories points up to a maximum of 13 pts.

Iteration 6, (1 story, 13 points)
During this iteration, we began work with the new stakeholders, ASTTeQ. We completed initial surveys regarding what their current needs are. Based off their most needed requirement, we began implementing a social worker website. This iteration we completed: setting up the Angular client, Express server, Mongo database and participant creation.

Iteration 7, (5 stories, 15 points)
For this iteration we worked on features that allow a social worker to help a participant find a housing solution. This includes adding housing resources, managing a participant's profile, and assigning a participant to housing.

Release 2 Total: stories, 46 points over 7 weeks
Release 2, Iteration 8, (4 stories, 18 points)
In this iteration we completed authentication for the users, adding notes related to participants, keeping track of participant logs, and creating new casefiles for participants.

Plan at least two iterations in advance

Iteration 9, (6 stories, 21 points)
Validation for forms 5pt
Add Documents 3pt
SocialWorker Account Privileges 3pt
SocialWorker Online Website 3pt
Enhanced Navigation 5pt

Assign Participant to Housing Form 3pt

Iteration 10, (5 stories, 21 points)

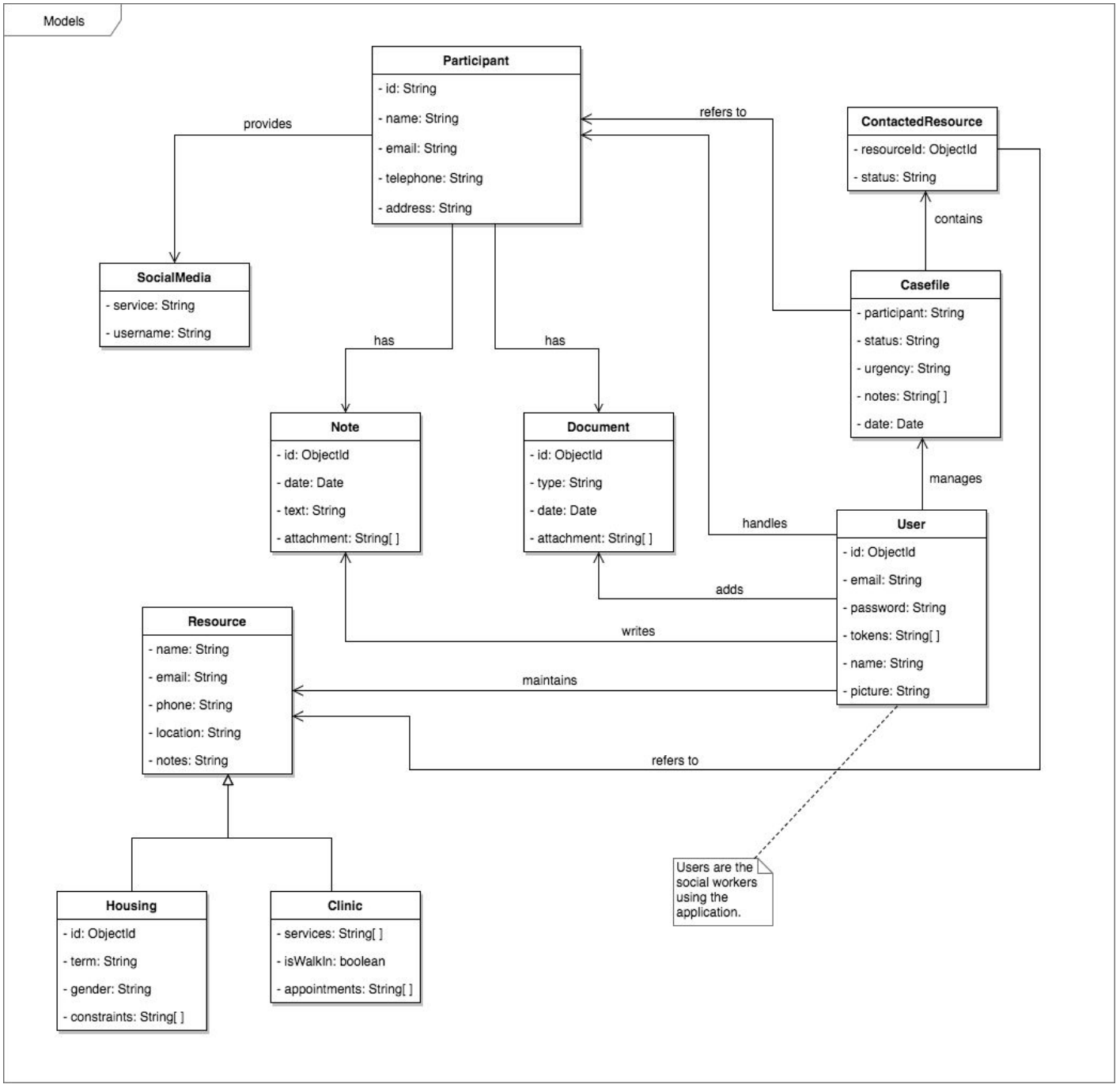
Continuous deployment 3pt
PDF viewing + markup 8pt
Dashboard Landing 5pt
Internationalization support 3pt
Custom and uniform UI theme 2pt

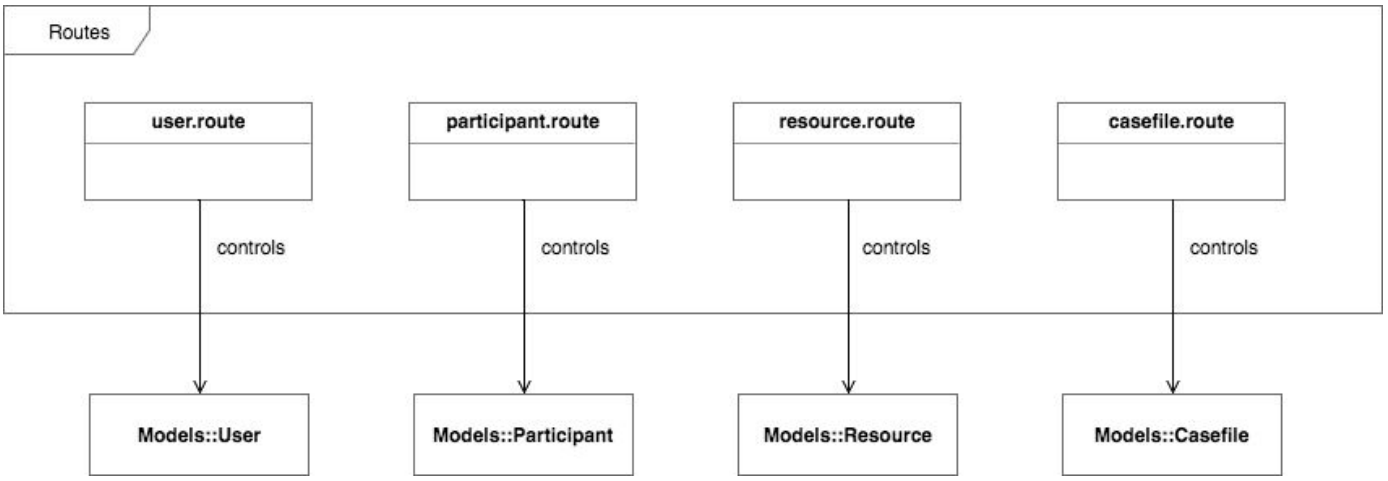
Max four sentence paragraph describing main achievements.

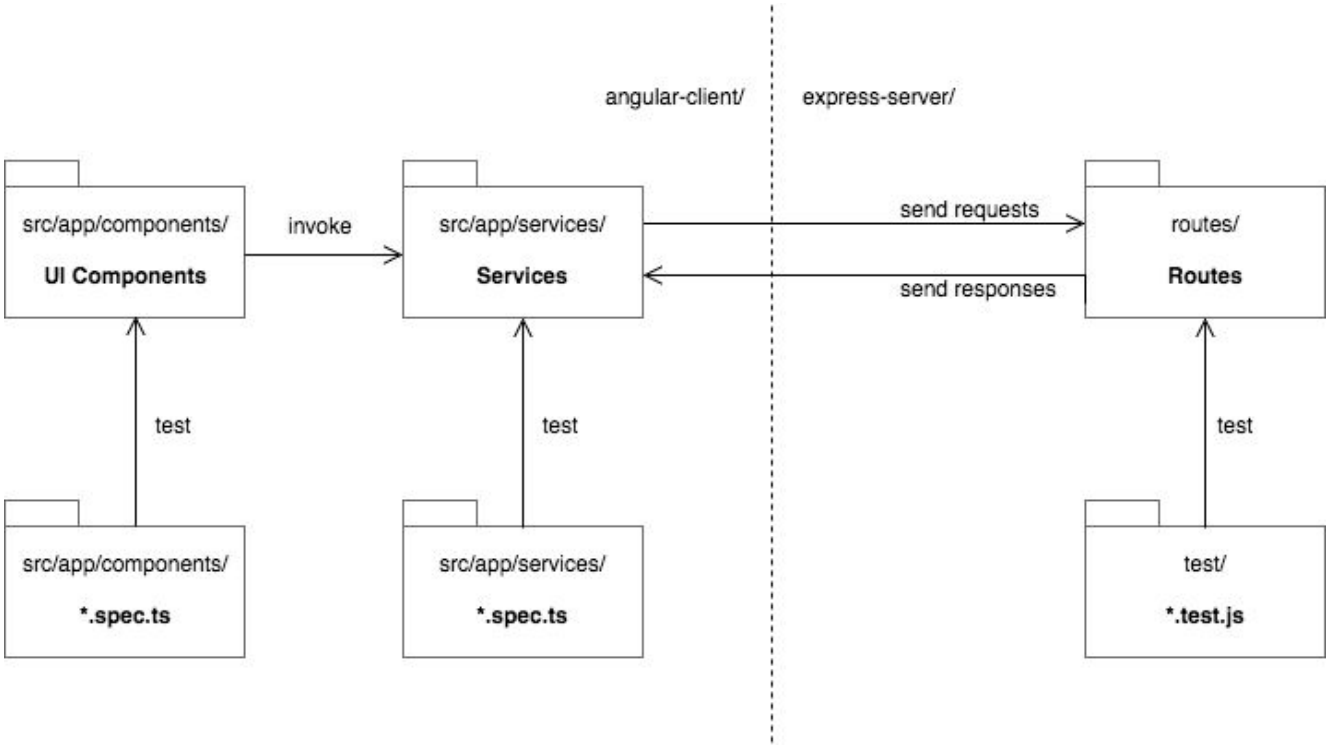
We have found a truly engaged stakeholder who gives us meaningful feedback and who we can brainstorm for the direction of our project. So far we have developed a proof-of-concept application that has the basic capabilities of management of casefiles for frontline social workers.

Overall Arch and Class diagram

Show us the layers in your system and your domain classes. You can also include individual class diagrams in your stories on GitHub


Models
Participant
- id: String
- name: String
- email: String
- telephone: String
- address: String
provides
refers to
ContactedResource
- resourceId: ObjectId
- status: String
contains
SocialMedia
- service: String
- username: String
Casefile
- participant: String
- status: String
- urgency: String
- notes: String[ ]
- date: Date
has
has
Note
- id: ObjectId
- date: Date
- text: String
- attachment: String[ ]
Document
- id: ObjectId
- type: String
- date: Date
- attachment: String[ ]
manages
handles
User
- id: ObjectId
- email: String
- password: String
- tokens: String[ ]
- name: String
- picture: String
adds
writes
Resource
- name: String
- email: String
- phone: String
- location: String
- notes: String
maintains
refers to
Housing
- id: ObjectId
- term: String
- gender: String
- constraints: String[ ]
Clinic
- services: String[ ]
- isWalkIn: boolean
- appointments: String[ ]
Users are the social workers using the application.


Routes
user.route
participant.route
resource.route
casefile.route
controls
controls
controls
controls
Models::User
Models::Participant
Models::Resource
Models::Casefile


angular-client/
express-server/
src/app/components/
UI Components
invoke
src/app/services/
Services
send requests
send responses
routes/
Routes
test
test
test
src/app/components/
*.spec.ts
src/app/services/
*.spec.ts
test/
*.test.js

Infrastructure

List of Frameworks:

- Angular5 https://angular.io/
- Node https://nodejs.org/en/
- Express https://expressjs.com
- MondoDB https://www.mongodb.com

Infrastructural Changes:

After switching to a new stakeholder, our application domain changed. We no longer use TensorFlow, Electron, with the current application. Therefore, the

Name Conventions

List your naming conventions or just provide a link to the standard ones used online.

Database: We will follow the MongoDB Style guide
Backend: We will follow the Microsoft TypeScript Style Guide
Frontend: We will follow the Angular Style Guide.
Frontend styling: We will follow the Sass Guidelines.

Code

Key files: top **5** most important files (full path). We will also be randomly checking the code quality of files. Please let us know if there are parts of the system that are stubs or are a prototype so we grade these accordingly.

File path with clickable GitHub link	Purpose (1 line description)
angular-client/src/app/components/participant-profile/participant-profile.component.ts	Handles participant management on the angular client.
angular-client/src/app/services/authentication.service.ts	Handles authentication (login, register heartbeat) in the angular client.
angular-client/src/app/components/case-modal/case-modal.component.ts	Handles the creation of a new case for a Participant
express-server/routes/participant.route.js	Handles requests for all participant data, interacts with the database.
angular-client/src/app/components/profiles/profiles.component.ts	Handles the viewing/ editing of Participants in the frontend.

Testing and Continuous Integration

Each story needs testing before it is complete. If some class/methods are missing unit tests, please describe why and how you are checking their quality. Please describe any unusually aspects of your testing approach.

List the **5** most important test with links below.

Test File path with clickable GitHub link	What is it testing (1 line description)
casefile.test.js	Testing that Casefiles are created and uploaded correctly in the database
participant.test.js	Testing that Participants are created and uploaded correctly in the database
resource.test.js	Testing that Resources are created and uploaded correctly in the database
user.test.js	Testing that Users are created and uploaded correctly in the database
participant-profile.component.spec.ts	Frontend testing for the component

We are using TravisCI, and the link is https://travis-ci.org/adriannadiaz/SocialWorker/builds/