

## COURSE OVERVIEW

**Course Title:** Level 6: Project for Social Good

**Course Number:**

**Number of Units:** 3.0 units

**Total Hours of Instruction:** 25 - 35 (depending on complexity)

### Course Description

As a capstone course, Level 6 offers students the opportunity to collaborate on diverse teams to develop a novel software solution that addresses a real-world need originating from local citizens or other non-profits. Students apply computational thinking practices, interdisciplinary knowledge, and professional skills as they work through the software design process. Effective practices in problem solving, implementation, presentation, and collaboration are central to the course.

### Content and Evaluation

League students are taught to code by industry professionals who volunteer their time weekly to pass on their craft to the next generation. In order to encourage students to carry this spirit of giving into their own lives as computer programmers, the project developed in this level must benefit society. Previous apps built in level 6 have addressed societal problems (obesity, heart disease) or helped other non-profits do their work more efficiently (food distribution).

The platforms used for these projects are typically Android or web, allowing the student to familiarize themselves with databases, web services, and front-end programming.

The projects are publicly published on GitHub and will form part of the student's professional portfolio of work.

Students work in teams to implement the project with the guidance of a professional software engineer who has at least 5 years of software industry experience. Through lectures and project-specific examples, the software engineer mentors the students in Agile/Lean methods, and industry best practices such as design patterns and unit testing.

We mirror a real-world development environment as closely as possible by using professional version control to facilitate team work, professional project management tools such as Trello, and interfacing with the customer to clarify requirements, deliver frequent releases, and receive regular feedback.

We also learn the Agile engineering practices of pair programming, unit testing, continuous integration and refactoring, through practical application.

To complete this course, students must deliver a fully tested application that satisfies the following requirements:

- Application is in use by end users
- For social good
- Developed collaboratively with other team members
- Android apps must be available on the Google Play app store
- Other apps must be open source and published on GitHub

## Extra Credit

n/a

## Methods of Instruction

- |                                     |                                    |                                     |                             |
|-------------------------------------|------------------------------------|-------------------------------------|-----------------------------|
| <input checked="" type="checkbox"/> | CLASS DISCUSSION/DISCUSSION BOARDS | <input checked="" type="checkbox"/> | LECTURES                    |
| <input type="checkbox"/>            | FIELD TRIPS                        | <input type="checkbox"/>            | CASE STUDIES                |
| <input checked="" type="checkbox"/> | GROUP WORK                         | <input checked="" type="checkbox"/> | OTHER: SOFTWARE DEVELOPMENT |

## Out of Class Assignments

Total hours expected to complete assignments: n/a

- |                                     |                    |                                     |                             |
|-------------------------------------|--------------------|-------------------------------------|-----------------------------|
| <input type="checkbox"/>            | TEXTBOOK EXERCISES | <input type="checkbox"/>            | READINGS                    |
| <input checked="" type="checkbox"/> | GROUP WORK         | <input type="checkbox"/>            | WRITTEN ASSIGNMENT/ESSAY(S) |
| <input checked="" type="checkbox"/> | STUDENT PROJECT    | <input checked="" type="checkbox"/> | OTHER: SOFTWARE DEVELOPMENT |

## Evaluation/ Grading

- |                                     |                             |                                     |                                       |
|-------------------------------------|-----------------------------|-------------------------------------|---------------------------------------|
| <input type="checkbox"/>            | EXAM(S)                     | <input type="checkbox"/>            | CLASS PARTICIPATION/DISCUSSION BOARDS |
| <input checked="" type="checkbox"/> | WRITTEN ASSIGNMENT/ESSAY(S) | <input checked="" type="checkbox"/> | CLASS PROJECT(S)                      |
| <input type="checkbox"/>            | OTHER:                      |                                     |                                       |

## Topical Outline

The technical topics covered in Level 6 are dependant on the technologies used for the application. As an example, topic #4 would be applicable for an Android app. In addition to learning the technology stack relevant to the project, we cover the more general topics outlined below.

### 1. Working in Teams

- Advanced Git: forking, merging, branching
- Pair programming
- Continuous integration
- Collective code ownership

### 2. Project Management

- Creating and managing a Kanban board
- Requirements gathering / user stories
- Project lifecycle
- Working with the customer
- Task estimation

### 3. Agile/Lean

- Minimum viable product
- Test driven development
- Refactoring
- Simple design

#### **4. [Example] Android Development**

- Android SDK
- Activity lifecycle
- Fragments & user interface elements
- Saving data
- Emulator
- Location & maps