

Web Development

Group Project Expectations

- Group Projects: 4-6 people per group
- Feature Requirements: 3 features per group member
- Total features per project: 12-18 features

Each member of the group is responsible for developing at minimum 3 features of the project. Learners may work collaboratively, however, they must be the main contributor for 3 features. LaunchCode encourages creativity in project design, and this document should be a guide for planning. If a feature is selected that is not listed below, TAs and the Learner must meet with the CEM to get their feature approved.

Web development projects are required to use **Java(Spring)/C#(.NET) and SQL**.

If learners are interested in utilizing a front end framework, we currently recommend React.

Must Include

- Full CRUD Functionality
- User Account Creation and Management
- SQL Database
- External API usage

Additional Features List

- Blog functionality
- Ability to favorite items or posts to a user account
- Ability to search the database
- Contact form
- Ability to leave comments
- Mobile Responsiveness
- Ability to purchase something on the website
- Ability to upload a photo to the site and apply tags to it
- Implement a review system
- Navigation bar that adapts to user accounts
- Ability to sign up for a rewards system

Additional Recommended Languages or Frameworks

- Vue.JS
- Node.JS
- Bootstrap / Tailwindcss
- AWS
- Docker
- Expressjs
- MongoDB
- Angular

Project Outline

Copy this portion into a Google Doc for your group to update as the project progresses.

Project Name - Movie Night? Flick Finder?

Project Description

- Our project will sort movies by metadata that is provided by the user. (i.e. if user only has certain streaming services we are able to recommend a movie based off of what the user has) The goal is to make the selection process easier for the user based on both review data that we gather from either APIs or raw data provided by companies like IMBD or Rotten Tomatoes .

Feature List

- Listing movies by streaming service that user provide
 - Check boxes can be used to represent service the movie is on
- User registration form
- User login form
- User logout buttons
- User update password form
- Allowing user to leave “personal reviews” which they can view and store locally
- Movie history
 - Create movie history list
 - Delete
 - Display movie list (view)
 - Update movie list
- Save for later
- Recommend based on user provided metadata
- Filtering by
 - Decade
 - Genre
 - Runtime
 - Pick for me (Editors' picks?)
- genre/ categoriesAuto prompt if user is idle on the site (countdown timer)
- Short quiz/survey to recommendation of movie



Tech Stack

- Backend - SpringBoot and Java 11
- Frontend - Angular (Bootstrap/SQL/HTML/Typescript/Javascript)
- Database - mySQL
- Movie API - maybe Watchmode

Group Members and Work Division

- Aaron Alexander
- Elizabeth Karr
- Erin Kacerovskis
- Harmony Lovell
- Nichelle Barton

Project Links

- GitHub: <https://github.com/KC-LCW-Jan-23-Classroom/liftoff-angular-in-the-outfield>
- Link to Project Management Platform:
<https://github.com/orgs/KC-LCW-Jan-23-Classroom/projects/1/views/1>
- Presentations
 - Group Presentation
 -  Group Project Presentation Template
 - Individual Presentations
 -  Individual Project Presentation Template

The project outline below is to give you an idea of what a Liftoff project can look like. It is not a project that learners can refer to on GitHub.

Sample Project Outline

Project Name - Get a Slice

Project Description

- Our project is an online ordering platform for a local pizza company. Our group found that the current format for online pizza ordering is missing features that could lead to a better user experience, so we developed a fake pizza company to showcase what we believe to be a better website.

Feature List

- **Customers can create and manage an account**
- **Managers can adjust menu items, keep track of inventory, and mark items as on special**
- **Menu items stored in an SQL database**
- **Utilizes Stripe API to take user payment information**
- Customers can favorite menu items
- Ability to sign up for a rewards system
- Ability to re-order previous carts
- Ability to use rewards points to purchase items
- Users accounts are verified through email
- Users can get in touch with the shop using the Contact Us form
- Users can order pizza or other items on the site
- Users can visit a blog to see new updates for the company
- Customers can leave comments on the blog

Tech Stack

- Java and Spring for back end
- React for front end
- SQL for Database
- Stripe API for payments

Group Members and Work Division

- Lexi
 - **Customers can create and manage an account**
 - Users accounts are verified through email
 - Customers can favorite menu items
 - Customers can leave a review for their order
- Melinda

- **Managers can adjust menu items, keep track of inventory, and mark items as on special**
 - Sign up for a rewards system
 - Users can get in touch with the shop using the Contact Us form
- Shaw
 - **Utilizes Stripe API to take user payment information**
 - Users can order pizza or other items on the site
 - Ability to use rewards points to purchase items
 - Ability to re-order previous carts
- Maggie
 - **Menu items stored in an SQL database**
 - Users can visit a blog to see new updates for the company
 - Customers can leave comments on the blog

Project Links

Github:

Group Presentation:

Individual Presentations

- Maggie
- Shaw
- Melinda
- Lexi

Project Management: (Utilize GitHub Projects)