BadgR Presentation Timeline (and pictures)

[bit.ly/BadgRPrezi](https://bit.ly/BadgRPrezi)

BadgR Source Code

<https://github.com/B1sth3be5t-1/BadgR>

BadgR Narrative

1. **Describe the project’s objectives, vision, and/or overall purpose. Why did you choose your topic? What kind of effect do you want to have on your audience?**
   * BadgR is an application designed to allow scouts to keep track of their merit badges in the easiest and cleanest way. Scouts are able to explore merit badges and their requirements, as well as keep track of what badges they are working on, and which ones they have already completed. Scoutmasters are able to see which scouts are in their troop, see newly completed merit badge notifications, and assign scouts certain merit badges. I chose this topic because as an Eagle Scout, I not only wanted to give back to my scouting community, but I also wanted to give other scouts a chance to keep track of their information in the most streamlined approach possible. I wanted to give scouts a clean, no-hassle, and portable resource to access their information. I want my audience to have an easy time navigating BadgR, as well as give them a digital resource for their Scouting journey.
2. **List the software applications used to create the project. List any additional equipment (scanner, digital camera, etc.) used in the creation of your project.**
   * I used the Android SDK (Software Development Kit) to write all of my code, and push it out to my device. I also used Pixlr E to create the BadgR icon and full image. I used Canva to design and print my poster board.
3. **Provide a brief description of the process used to create the project including the amount of time spent creating it and the names of all contributors.**
   * I started by asking a fellow Eagle Scout what he would like to see in a merit badge tracker app, and took notes of his responses. I knew what I wanted to implement with my app, but I wanted to give myself a range of ideas to see what my fellow app users wanted. I started by creating a TODO list of all of the key ideas and processes of the app that I wanted, and then split them down into smaller lists to complete (such as the file layout, elements, and backend processes). Before I started writing any code, I made myself flowcharts (shown in my process Prezi) and drew out how I wanted the app, connections, and logic flow to work. Once I reviewed this with my advisor, I began to throw myself into the project over Thanksgiving break, and began checking off items as I completed them. I have been working on the project for 7 months, and have put in around 150 hours.
   * Resources:
     1. Austin Myers - Cousin, helped me get MySQL database set up on the computer. Also reviewed flowcharts
     2. Billie Jo Scott - Friend from Scouts BSA local office, got copyright information about the project and resources
     3. Ben Stoudt - Eagle Scout, helped with ideas for features of the app
     4. Mr. Fleming - Numerous hours of listening to my rants about ideas, steered me in the right directions when I got too excited over a particular feature I was working on
     5. Mr. Lynch - Printed presentation poster
4. **List any templates you may have used, altered, or modeled your project after, and describe the changes you made to adapt it to your project (if you did not use any templates, put Not Applicable).**
   * When starting this project, I had never done any Android development. I started with the Android SDK Login and Register Screen templates. I began to analyze code to understand how Android processes work. I got these pages working with the MySQL database on my computer, but largely left it alone to start. As I continued through the project and learned how to program to suit an Android application, I often went back and reworked the login and register page to make it suit my app better. By the end, I had reworked the theme, and changed all of the original code to fit my style of programming and logic.
   * I used a YouTube video to learn how to create tabs in my project using a fragmentPagerAdapter. I would have used android examples, however there was one method of creating these files that was deprecated (meaning it is not currently supported by android) so I needed to find a method that was supported, and there were no official android tutorials available. ([fragmentPagerAdapter deprecated android | tablayout in android studio | view pager 2 | Tech Projects](https://www.youtube.com/watch?v=rbF_KqH5oS8&list=WL&index=13))
   * I also used an expandable list for **“Search”** and **“My Badges”** tabs, however I needed to find a tutorial for a CustomExpandableListAdapter, which instead of creating other text boxes when an element is expanded, creates a template view with pictures, text, and a checkbox. <https://www.geeksforgeeks.org/simpleexpandablelistadapter-in-android-with-example/>
5. **Cite ALL sources of information, audio, video, images, etc. used in your project, including your own work. For any element that is not your own original work and/or owned by someone else, written permission from the copyright owner is REQUIRED and should be attached.**
   * Badger picture:
     1. Badger drawing - Hand drawn by friend, scanned into Pixlr E
     2. Badge sash - drawn in Pixlr
     3. Badges - “Clipart, Vectors and Royalty Free Photos at Clipart.com School Edition.” Clipart.com School Edition, https://schools.clipart.com/
     4. Merit badge icons - “Prepared. for Life.™.” *Boy Scouts of America*, 8 Feb. 2022, <https://www.scouting.org/> (I contacted Billie Jo Scott at my local scout office (see resources from #3) and she said that all of the badge icons and information are not copyrighted material, so I am free to use them in my project)
6. **Provide any additional information the judges should be aware of while evaluating the project.** 
   * As mentioned in #5, all of my pictures, badge icons, and information are all not copyrighted, so I did not need permission forms to use these materials.
   * The app looks simple and clean, as the majority of my time was spent in the backend of the project. For any judge who knows about programming from scratch, this took a very long time and has many moving parts, so I am definitely proud of what I have created.
7. **If necessary, provide any information judges will need to know in order to operate/understand the program, or provide a step-by-step tutorial. Please make sure all passwords are removed from the computers to ensure the judges are able to access the projects.**
   * Steps are on my poster, look for the bolded instructions under the story section.

Pseudocode

* Open Main Activity
  + Initial database connection is made to pull all requirements for all merit badges in the database. Allows for quicker and more convenient access in the scout page.
  + Display Elements (Image, Buttons for login and register)
  + Click on Login (Find this color)
  + Click on register (Find this color)
* Login:
  + User inputs username and password
  + If both username and password valid, activate submit button
  + User clicks submit button
  + If email not in database
    - Display “Username not found. Please go to Register page”
    - User clicks “Don’t have an account?” text
      * Goes to Register Page
  + If email in database
    - If password is incorrect
      * Display “Incorrect password”
    - If password correct
      * Open Scout page (Find this color)
* Register:
  + User inputs First & last name
  + User inputs email and password
  + User inputs age and troop
  + User clicks submit
    - If email already in database
      * Display username already in database
    - If register successful
      * Display register success message
      * Open login screen
  + User clicks “Already have an account?” text
    - Open Login screen
* Scout Page
  + Initialize 4 tabs of the scout page. Make initial database connections so that data loads faster
  + **Welcome** tab
    - Get number of badges in progress, completed, and eagle required badges in progress from database
    - Display in table with graphic and welcome
  + **Search** tab
    - Displays elements
    - User enters the badge they want to search for
      * Database search, display list
      * Badge clicked on
        + Open badge view
        + Add or remove badgeID from list if box is checked or unchecked
    - Submit button clicked
      * All badges in the list are added in the database
  + **My Badge List** tab
    - Database connection is made to pull all badges in progress
    - List is created with merit badge names and progress bars
    - Merit badge clicked
      * Opens up a sub-list with all of the requirements with a checkbox
      * Checkbox checked or unchecked
        + Requirement number and badgeID added or removed from checked list
    - Submit button clicked
      * Database connection is made and all requirement changes updated
      * If any badges are fully completed
        + Badge is removed from “My Badges” list, and added to completed list
  + **Completed badges** tab
    - Makes database connection to pull completed badges
    - List is created
    - Edit button clicked
      * Click checkbox next to badge they wish to remove
      * Click submit
        + Badge removed and added to “My Badge List” with all requirements completed
* Scoutmaster Page
  + Initialize 4 tabs of the page, makes initial database connections so that data loads faster
  + **Welcome** tab
    - Get number of badges of the troop in progress, completed, and eagle required badges in progress from database
    - Display in table with graphic and welcome
  + **Notifications** tab
    - Makes database connection to get notifications, display new ones in list
    - Clear button clicked
      * All new notifications are cleared
    - Previous button clicked
      * Displays all previous notifications
  + **My Scouts** tab
    - Display list of all scouts in troop
    - Click on profile button
      * Opens up a new screen showing the scout’s badges in progress, their percentage complete, and completed badges
  + **Search for badges** tab
    - User enters merit badge to search for
    - Database connection made, pulls all badges and creates list
    - Clicks on checkbox
      * Assign button activated, added to list
    - Clicks on Assign button
      * Sub list displayed, with all scouts in troop
      * Box checked
        + Scout added to list
      * Submit button clicked
        + All badges are added to all scouts checked