

Project spec: LookOnPoint

Group members: Jonathan Rintala

Project Description:

- Working name of project: LookOnPoint
 - The main goal of the project is to deliver a mobile app for sharing and rating clothing styles in a community format via images; delivering a peer-to-peer platform to acquire honest opinions about personal everyday fashion, helping people to decide what to wear for all types of occasions as well as for consuming content and getting inspiration. The users should furthermore have the ability to interact with one another in real time, through a comment field attached to each look uploaded on the platform.
- The API will consist of a number of endpoints to ensure people are able to upload pictures/posts, create profiles and interact with each other by commenting and liking. The admins should also have possibility to handle and manage the users.
 - /api/addUser
 - param: user to add
 - /api/removeUser
 - param: user to remove
 - /api/getAllUsers
 - /api/getUser
 - param: user to get
 - /api/addImage
 - /api/addCommentOnLook
 - /api/addLikeOnLook
 - /api/getLookInfo
 - /api/addLook

Database Description:

- The database will contain
 - TypesOfLooks:
 - Id: String (PK)
 - Name: String, predefined categories - formal, casual, beach, etc.
 - Users:
 - Id: String (PK)
 - UserName: String

- Status: String
- Age: Int
- Gender: "M"/"F"
- AvgRating: Float
- Description: String
- Looks:
 - Id: String (PK)
 - LookName: String
 - LookType: String, FK Id.TypesOfLooks
 - LookImage: String, FK
 - UploadedByUser: String, FK Id.Users
- Comments:
 - UserCommenting: FK Id.Users
 - LookCommentedOn: FK Id.Looks
 - Comment: String
- Likes:
 - UserLiking: FK Id.Users
 - LookLiked: FK Id.Looks

Grade Aim:

- A/B

Stack Description:

- Frontend: React Native, with Redux
- Backend: Django, MongoDB or PostgreSQL

Views:

- Login/signup
- My profile
 - Possibly my recent looks, some overall score, achievements, etc.
- Upload new look - LookOnPoint?
 - Let user upload picture of their style/look and add a comment
 - Allow for user to add "type" of look - casual, formal, beach, etc.
- Feed of styles/looks
 - Ability to interact through votes and comments
 - Possibly sortable on fashion category, gender, region, etc.
- Comment
 - The commenting view where people will be able to discuss and leave honest opinions on the looks in real time