Hinder
Tinder for Hackathons

Discussion 1B
Team Name: TBD

Kimberly Svatos, 604425426
Marshall Briggs, 304417630
Kyle Haacker, 904467146
George Archbold, 604407413
Daniel Berestov, 404441309
Apurva Panse, 504488023

**Repository:** https://github.com/marshdevs/Hinder

Project Motivation

- Forming teams for hackathons, class projects can be difficult, awkward
- Our goal: help people build better groups, faster
  - Make it easy for users to browse events and projects that interest them
 - Help groups select the best candidate based on experience, skillset

Hinder

Overview / Application Dashboard

- Event planners…
  - Create events
  - Edit event information (name, date, location, etc.)
- Event attendees…
  - Create profiles, list their skillsets
  - Browse and sign up for events
  - Search for projects to join (or create a project)
  - Manage their events and projects
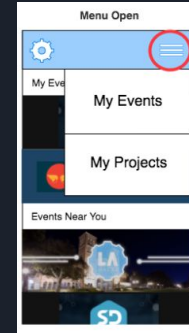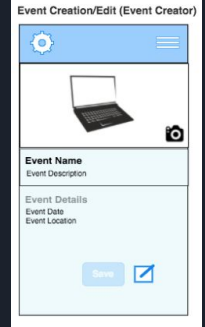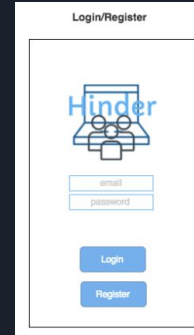
User Profile

- Hinder users can update their profile at any time:
    - Profile picture
    - School or company
    - Experience level
    - Skillset
- Groups leverage this information to pick the candidate with the best background, experience level, and skillset for the project
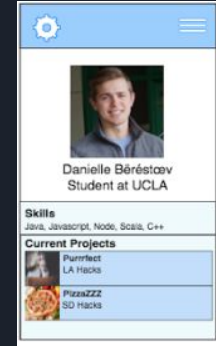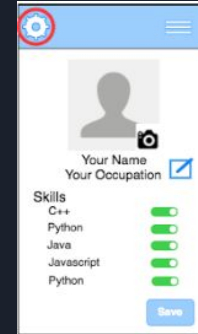
Project Creation and Description

- Users can browse and enroll in several events
  - Allowed to join one group project per event
- Alternative to joining a group - create a new project!
  - Specify the project name
  - Add a cover photo
  - Designate group size
  - Describe project goal and desired member skills
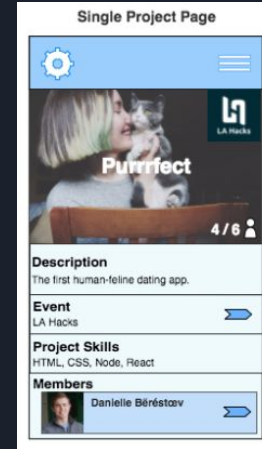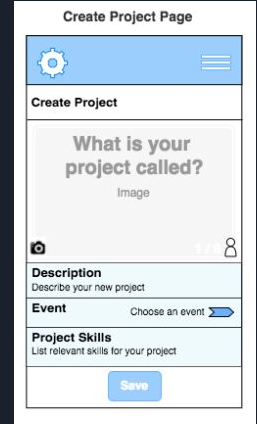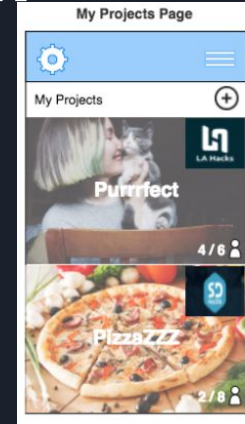
Project and User Interaction

- How do project managers fill their team with skilled members?
- How do event attendees find a project that suits their interests?
- Swiping interface - similar to Tinder
  - Project managers swipe left on users with desired skillset
  - Users swipe left on projects that interest them

Search Users Page
(Group owner)

Danielle Bëréstœv
Student at UCLA

Skills: Java, Javascript, Node, Scala, C++

Showing users for: LA Hacks

Search Projects Page
(Group member)

Purrrfect
The first human-feline dating app.

Members: 4 / 6
Needs Skills: HTML, CSS, Node, React

Showing groups for: LA Hacks

Hinder
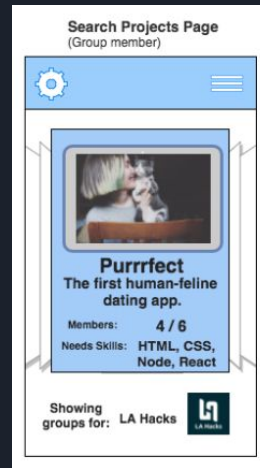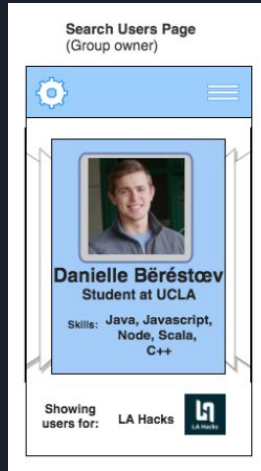
Mediator Design Pattern

- Users have potential to interact with dozens of different teams
- Teams could interact with hundreds of potential members at any given time
- Each interaction could be one of several different types
    - User swiped right on team
    - Team swiped left on user
    - Team and user both swiped right on each other
- It would not be feasible to store *everyone's* interactions client-side.
- Interactions are stored in a database, and updated as user->project relationship changes
- Users and projects interact with their event's Mediator, which monitors that relationship
- The Mediator automatically adds user to group if a match is made

Class Diagram

Event Attendee

- enrolledEvents: HashMap<Event, Project>
- skillSet: SkillSet
+ experience: int
+ year: int
+ school: name

+ setProject(Event, Project): bool

Event Planner

+ events: ArrayList<Event>

- deleteEvent(Event) : bool
- createEvent() : bool

User

- name: string
- id: int
- skillSet : SkillSet

SkillSet

- cpp: bool
- java: bool
- c: bool
- python: bool
- ios: bool
- javascript: bool
- HTML/CSS: bool

- setSkill(string, bool): void

Project

- Project(User, Event) : Project
- id: int
- manager: User
- users: ArrayList<User>
- event: Event
- openStatus: bool

- setOpenStatus(bool): void
- getOpenStatus() : bool
- addUser(User): void
- removeUser(): void

Event

- id: int
- description: string
- name: string
- users: ArrayList<Users>
- projects: ArrayList<Project>
- location: string
- mediator: Mediator

+ addUserToPool(User): bool
+

Mediator

+ field: type

+ userInitSwipe(User, Project, bool): void
+ projectInitSwipe(User, Project, bool) : void

Application

+ events: ArrayList<Events>

+ createProfile(): bool

Use Case Diagram

Feasibility

- Hinder is a service that no other current application provides
- Current alternatives: word-of-mouth, matching events, Piazza
  - Less effective, efficient methods of exchanging experience levels and skillsets
- MVP has been narrowed down to specific set of key features
  - Team possesses the experience with Objective-C, Swift to implement this features
- Features can be primarily implemented using tools internal to Swift
  - Minimal dependency on unreliable APIs
  - Apple Core Location framework makes location-related features a possibility

Capability

- What makes us feel up to the task?
  - Extensive internship experience: Amazon, Facebook, USG, ITG
  - Strong familiarity and capability in front- and back-end
  - Additional experience from school, club projects
  - Strong familiarity with Objective-C and Swift - past the learning curve
  - Have extensively researched the APIs, tools needed to create Hinder

Hinder

Questions?

Hinder