BIZREACH

Where Service Meets You, Anywhere

By: Anas, Amjad, Harshil, Aneeq, Majed

What Problems does BizReach Solve?

BizReach is a platform for individuals above the age of 16.

Provides clientele for Business owners and services for customers.

Main Solutions

- Trust and Credibility
- Convenience
- Local Business Promotion

Process

- All Communication: Discord
- Initial Meetings at start of sprint
- GitHub Workflow
- Good decisions: Detailing workflow, Cleaning up code base
- Bad decisions: Neglecting UI

Technical Discussion - FERN Stack

Technical Items:

Chakra UI

Significant difference in appealing UI

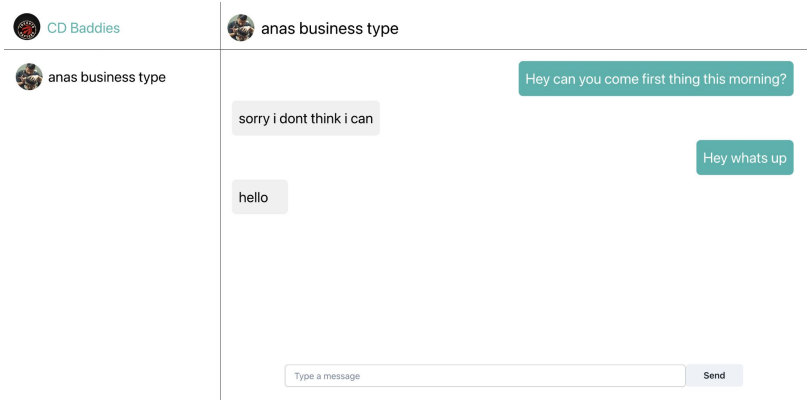
Socket.io

Enabling bi-directional communication between clients and businesses

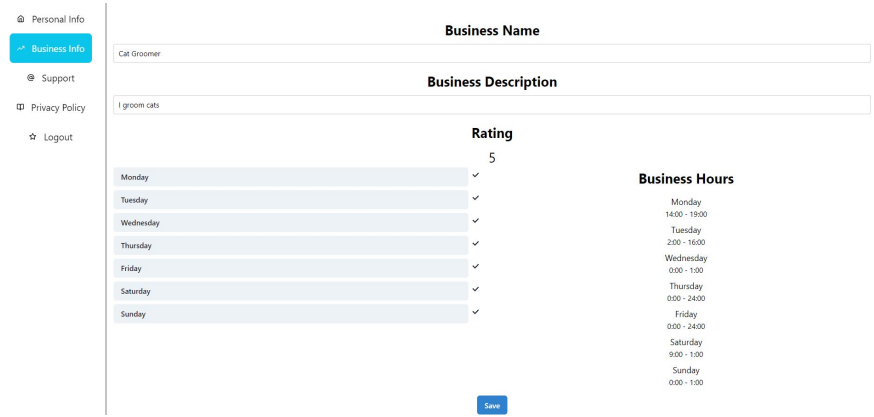
Firebase

Real-time data synchronization for chats, posts & accounts

Technical Discussions




```
useEffect(() => {
  const socket = io('http://localhost:3000');
  socket.on('newMessage', (data) => {
    setChatMessagesData((prevChatMessages) => ({
      ...prevChatMessages,
      [data.senderId]: [
        ...(prevChatMessages[data.senderId] || []),
        { text: data.text, sender: { _path: { segments: ['', data.senderId] } } },
      ],
    }));
  });
  return () => {
    socket.disconnect();
  };
}, []);
```



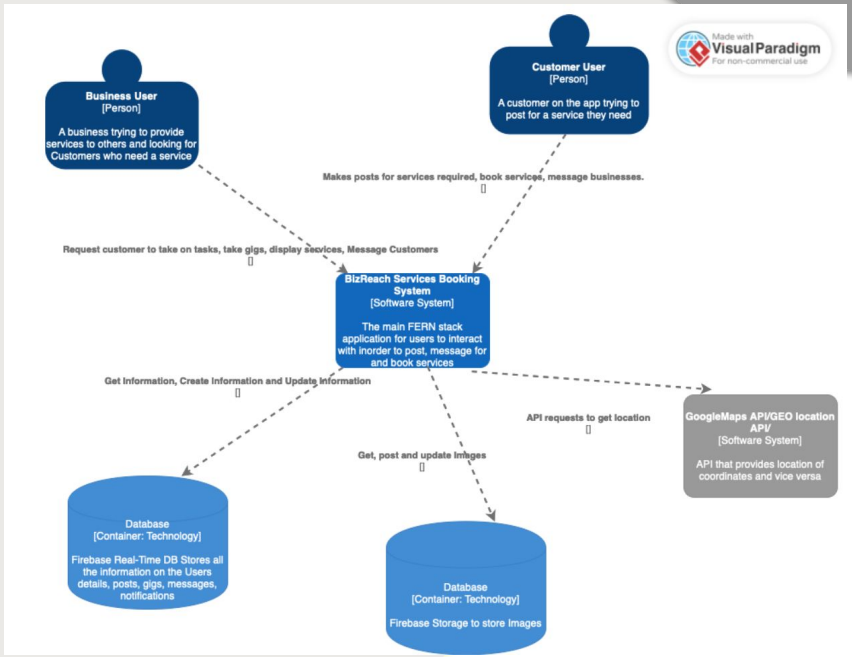
Software Architecture

- Main components
- Responsibilities
  - Interaction
- Technology and tools used
- Challenges faced
- Software techniques used

Software Architecture

Main components/Responsibilities and connection

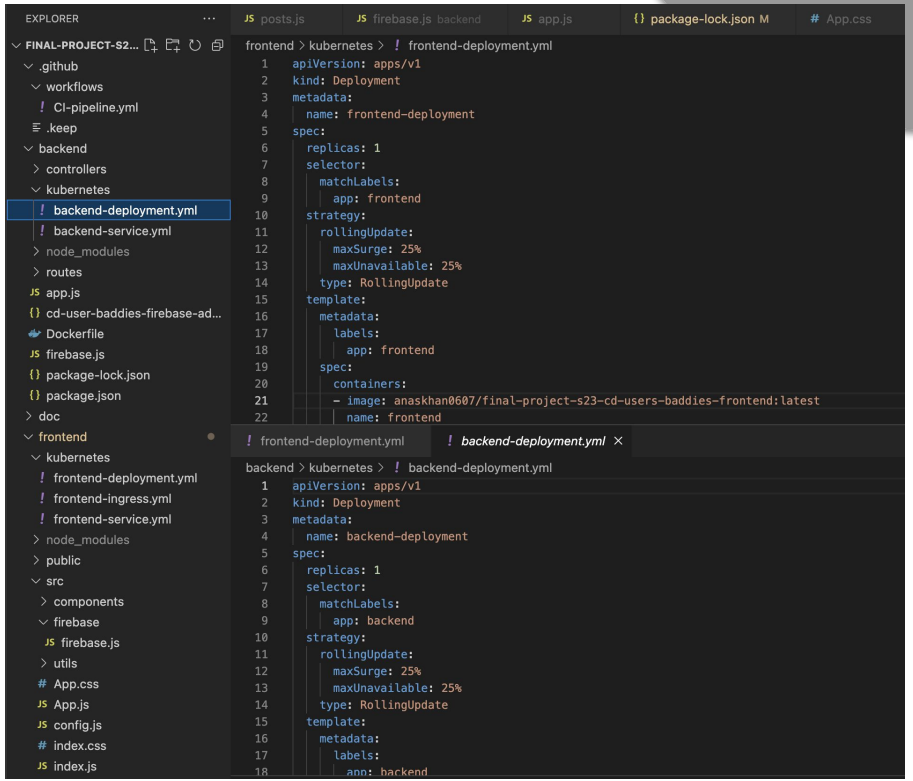
- React Frontend:
  - Intuitive user interface
  - Dynamic user experience
- Express.js Backend:
  - Intermediary between frontend and Firebase
  - Handles business logic and functionalities
- Node.js:
  - Powers the server environment
  - Efficient communication between frontend and backend
- Firebase:
  - Real-time database
  - Authentication and user management
  - Image storage
- Google Maps API:
  - Translates user-provided locations into actionable data



Software Architecture

Deployment

- Docker:
    - Created Docker files for backend and frontend
- Kubernetes:
    - Set up deployments and services
- Ingress:
    - Configured for frontend
- EasyDNS:
    - Host obtained for frontend
- Google Kubernetes Engine (GKE):
    - Hosting platform for scalability and availability



Software Architecture

Technology and Tools Used

- Firebase:
    - Real-time database and authentication
- Express.js and Node.js:
    - Backend communication
- React:
    - User interface development
- Docker:
    - Streamlined deployment
- Kubernetes:
    - Container orchestration
- GKE:
    - Hosting and scalability
- Socket.io:
    - Real-time communication implementation

Software Architecture

Challenges Faced

- Challenge: Real-time updates for posts and messages
- Solution: Implemented Socket.io for persistent connections

- Challenge: Restructuring codebase for better separation
- Solution: Enhanced separation of frontend and backend for maintainability

Software Architecture

Conclusion

- FERN stack choice aligned with project needs
- Components worked together seamlessly
- Deployment process ensured scalability and reliability
- Overcame challenges using innovative techniques
- Excited to demonstrate our application's architecture in action

Each Members' Contribution

- Anas: posts gigs flow, structuring code
- Amjad: Sign up, registration flow, profile page
- Harshil: Messaging feature, uploading pictures feature
- Aneeq: Reviews, Posts and Business features
- Majad: Discover Businesses, my gigs

Q & A