**Project Summary: Understand**

**Tagline**

*"From Diagnosis to Understanding"*

**Objective**

Understand is a platform designed to provide comprehensive resources and community support for individuals diagnosed with various conditions later in life. The platform’s primary goal is to offer curated learning and support resources for each condition, supplemented by user-generated ratings to help highlight the most valuable and relevant content. The project also aims to eventually include features for user interaction, such as a blog for personal stories and possibly a forum for discussions.

**Target Audience**

* **Primary**: Individuals diagnosed later in life with conditions such as Autism, ADHD, Bipolar Disorder, and significant physical changes, along with their families and caregivers.
* **Secondary**: Health professionals and educators seeking to provide resources and support for these individuals.

**Core Features**

1. **Resource Directory**:
   * **Organized by Condition**: Each condition will have a dedicated page that lists external links to valuable learning and support resources.
   * **User Ratings**: Registered users can rate the usefulness of each resource, helping others identify the most relevant and helpful material.
2. **User Authentication**:
   * **Managed via AWS Cognito**: Ensures secure access to rating features and future content contributions, such as blog posts or forum discussions.
3. **Content Management**:
   * **Resource Links**: Administrators can easily add and update external resource links to keep the content fresh and relevant.
   * **Ratings**: User ratings are averaged and displayed alongside resources to guide other users.
4. **Future Enhancements**:
   * **Blog for Personal Stories**: Allow users to share their experiences related to specific conditions, providing a platform for connection and support.
   * **Forum for Discussions**: A more interactive feature where users can create threads, ask questions, and engage with others in the community.

**Technical Implementation**

* **Front-End**:
  + Built with React, providing a responsive and user-friendly interface.
  + Hosted on AWS Amplify for seamless deployment and continuous integration.
* **Back-End**:
  + **Database**: AWS RDS with PostgreSQL for managing structured data, including user accounts, resource links, ratings, and (in the future) blog/forum content.
  + **API Management**: AWS API Gateway and Lambda functions to handle API requests between the front-end and database.
* **Scalability**:
  + The platform is designed to scale as the user base grows, with the ability to integrate more features over time, such as personalized content recommendations and video hosting.

**Project Scope**

* **In-Scope**:
  + Launch the platform with the core feature of a resource directory and user ratings.
  + Basic user authentication to support ratings and future contributions.
* **Out-of-Scope** (for the initial launch):
  + Advanced features like forums and blogs, which will be considered for future development based on user demand and platform growth.
  + Hosting of video content, though external video integration (e.g., YouTube) will be supported.

**Milestones**

1. **MVP Development**: Focus on building the resource directory with user ratings and basic authentication.
2. **Beta Testing**: Release the MVP to a select group for feedback and refinement.
3. **Full Launch**: Publicly launch the platform with core features, with plans for future enhancements.

**Next Steps**

1. **Design Documentation**: Finalize the system architecture and database schema.
2. **Scope Document**: Detail the project’s deliverables, timelines, and resource allocation.
3. **Development Planning**: Plan the development phases, focusing first on the resource directory and user ratings.