## **SkillCentrum Backend Development Roadmap**

### **Technology Stack**

| **Layer** | **Tech** |
| --- | --- |
| Language | JavaScript (TypeScript) |
| Framework | NestJS |
| Database | MySQL |
| ORM | TypeORM |
| External API | Ceresense Portal |

## **Database Design & Architecture (Sprint 1 – 3 Days)**

### **Objectives:**

* Design the database schema based on entities
* Implement and seed careers and skills tables with default data

### **Entities:**

* users(id, name, email, password, phone, address, verifyDate, userNo, careerId, country, language, stage, photo)
* working\_experiences(id, userId, companyName, position, startDate, endDate, task)
* educations(id, userId, schoolName, startDate, endDate, course)
* certifications(id, userId, name, date, description)
* portfolios(id, description, photos, startDate, endDate)
* services(id, userId, summary, description, photos)
* service\_skills(serviceId, skillId)
* careers(id, name)
* skills(id, name)

### **Deliverables:**

* SQL schema or TypeORM entity models
* TypeORM seeders for careers and skills tables

## **User Registration & Onboarding (Sprint 2 – 1 Week)**

### **User Story:**

A user enters their AppNo/Admission No. The backend fetches data from Ceresense. If found, the user proceeds with registration.

### **API Endpoints:**

1. GET /auth/search-user?appNo=XXXX – Search user from Ceresense
2. POST /auth/register – Finalize user registration

### **Deliverables:**

* Integration with Ceresense Portal (via HTTP client module)
* User data persistence logic
* Error handling for missing records
* Validation and transformation layer

## **User Authentication (Sprint 3 – 2 Days)**

### **User Story:**

Users log in with their email and password for authenticated access.

### **API Endpoint:**

* POST /auth/login – Authenticate user and issue JWT token

### **Deliverables:**

* JWT-based authentication
* Password hashing using bcrypt
* Auth middleware/guards

## **User Profile Management (Sprint 4 – 1 Week)**

### **User Story:**

After logging in, users can complete their profile step-by-step:

1. Biodata
2. Working Experience
3. Education
4. Certification
5. potifolio

### **Protected Routes:**

* POST /profile/biodata
* POST /profile/work-experience
* POST /profile/education
* POST /profile/certification
* POST /profile/portfolio

### **Deliverables:**

* Authenticated endpoints
* Step-wise form submission support
* Entity validation and relational mapping
* Track stage to monitor progress

## **Home Page Data (Sprint 5 – 2 Days)**

### **User Story:**

Show all available careers. When a career is selected, list 20 random talents related to it.

### **API Endpoints:**

* GET /careers – Fetch all careers
* GET /careers/:id/talents – Fetch 20 random talents in that career

### **Deliverables:**

* QueryBuilder-based talent filtering logic
* Pagination/limit support

## **Talent Profile (Sprint 6 – 2 Days)**

### **User Story:**

Display full profile of a talent including experience, education, portfolio, and services.

### **API Endpoint:**

* GET /talents/:id – Fetch full profile of a talent

### **Deliverables:**

* Nested relationships (User → Experience, Education, Portfolio, Service)
* Optimized query with joins

## **Service Details Page (Sprint 7 – 2 Days)**

### **User Story:**

View service details with associated skills and user summary.

### **API Endpoint:**

* GET /services/:id – Fetch full service details with skills and user summary

### **Deliverables:**

* Composite response with user info + skills + service data
* Transform responses for frontend

## **Send Proposal Feature (Sprint 8 – 4 Days)**

### **User Story:**

Clients can upload a proposal and send it to the selected talent.

### **API Endpoint:**

* POST /proposals/send – Upload proposal and email to the user

### **Deliverables:**

* File upload handler (Multer)
* Email dispatch integration (SMTP/Nodemailer)
* Security: verify sender and sanitize input

## **Final Testing & QA (Sprint 9 – 2 Days)**

### **Tasks:**

* Full end-to-end testing of all features
* Seed/test user accounts and flow
* Fix any validation, performance, or logic issues
* Ensure role-based access and secure endpoints

## **Deployment & DevOps (Sprint 10 – 2 Days)**

### **Tasks:**

* Dockerize backend application
* Configure environment variables
* Deploy to server (e.g., DigitalOcean, AWS, or Railway)
* Set up CI/CD pipeline (GitHub Actions or similar)

## **Summary Timeline**

| **Sprint** | **Duration** |
| --- | --- |
| Database Design | 3 Days |
| User Registration | 1 Week |
| User Authentication | 2 Days |
| Profile Management | 1 Week |
| Home Page Data | 2 Days |
| Talent Profile | 2 Days |
| Service Details | 2 Days |
| Send Proposal | 4 Days |
| Final Testing | 2 Days |
| Deployment | 2 Days |
| Total | ~4 Weeks |