

Sure! Here's the formatted to-do list in Markdown:

---

## To-Do List To-Do List by Issue Type

### Code Review Code Review

#### General Code Quality

- ☐ Remove unused files (e.g., `bottler.py`, `carts.py`) – *Dhvani Goel*
- ☐ Add return type hints to route functions (e.g., `-> List[Course]`) – *Madison Lopez*
- ☐ Add logging and comments to logic-heavy parts of the planner – *Madison Lopez*

#### Schema & Model Use

- ☐ Use `response_model=...` in all endpoints – *Briana Lonappan, Dhvani Goel*
- ☐ Replace manually built dicts with Pydantic model instances – *Dhvani Goel, Briana Lonappan*
- ☐ Validate email field using `EmailStr` – *Dhvani Goel*

#### Validation & Reusability

- ☐ Restrict `grade`, `quarter_taken` fields with enums – *Uriel, Dhvani Goel, Madison Lopez*
- ☐ Create helper functions (e.g., `validate_student()`) to avoid duplication – *Briana Lonappan, Dhvani Goel*
- ☐ Split `get_student` into separate routes for planned and completed courses – *Uriel, Dhvani Goel*

---

## Schema/API Design Schema / API Design

### Route Naming & Structure

- ☐ Rename `/login/{username}` → `/login/{email}` – *Briana Lonappan, Dhvani Goel, Uriel*
- ☐ Move `/plan_course` and `/mark_course_completed` under `/students/{id}/...` – *All three authors*
- ☐ Rename `/courses/major/{id}` → `/majors/{id}/courses` – *Dhvani Goel*
- ☐ Remove test route `/hi` – *Dhvani Goel, Uriel*

## Data Modeling Improvements

- ☐ Convert `department_code` to Enum or use lookup table – *Madison Lopez*
- ☐ Add constraints for fields like `planned_quarter`, `grades`, etc. – *All authors*
- ☐ Use composite keys (`student_id`, `course_id`) for course records – *Madison Lopez*
- ☐ Use consistent model inheritance for cleaner schema (`CourseBase`, etc.) – *Briana Lonappan*

## Validation / Security

- ☐ Use POST with JSON body for login instead of path/query params – *All authors*
  - ☐ Ensure passwords are hashed (e.g., bcrypt) – *Briana Lonappan, Madison Lopez*
  - ☐ Add index on `email` field for faster logins – *Madison Lopez*
- 

## Test Results/Bugs Test Results & Bugs

### Planning / Enrollment Logic

- ☐ Prevent planning of a course that is already completed – *Dhvani Goel*
- ☐ Validate `major_id` and `course_id` to avoid server errors – *Briana Lonappan, Dhvani Goel*
- ☐ Normalize casing of department codes (e.g., accept `csc`, `CSC`) – *All authors*
- ☐ Restrict `grade`, `quarter_taken` to valid values only – *Uriel, Dhvani Goel, Madison Lopez*

### Endpoint Failures

- ☐ Handle 404s for invalid majors/courses with clear error messages – *All authors*
  - ☐ Avoid 500 errors during student creation with invalid major – *Dhvani Goel*
- 

## Product Ideas Product Ideas

### Graduation & Course Planning

- ☐ **Graduation Audit** – GET `/students/{id}/grad_status` – *Uriel*
- ☐ **Custom Planner** – POST `/planner/custom_plan` – *Dhvani Goel*
- ☐ **Optimized Path to Graduation** – GET `/planner/optimized_path_to_graduation` – *Madison Lopez*
- ☐ **Earliest Graduation Prediction** – POST `/students/{id}/earliest_graduation` – *Dhvani Goel*

- **Schedule Conflict Checker** – POST `/schedule/conflict_checker` – *Madison Lopez*

#### **Additional Helpful Endpoints**

- View students in same major – GET `/majors/{id}/students` – *Uriel*
- Degree audit overview – GET `/students/{id}/degree_audit` – *Dhvani Goel*
- Student schedule summary – GET `/students/{id}/schedule` – *Briana Lonappan*
- List popular majors – GET `/majors/popularity` – *Briana Lonappan*

---

Let me know if you want this exported to a file or integrated into an issue tracker.