

**XraysController**

«primary» context: HospitalDbContext  
«primary» mapper: IMapper  
- \_validator: XrayValidator = new() {readOnly  
}

+ «async» GetAll (): Task<IActionResult>  
    {Annotation = HttpGet}  
+ «async» GetById (id : int): Task<IAction  
Result>  
    {Annotation = HttpGet("{id}")}  
+ «async» Create (xrayAdd : XrayAddDto):  
Task<IActionResult>  
    {Annotation = HttpPost}  
+ «async» Update (xrayUpdate : XrayAdd  
Dto, id : int): Task<IActionResult>  
    {Annotation = HttpPut("{id}")}

**DiseasesController**

«primary» context: HospitalDbContext  
«primary» mapper: IMapper  
- \_validator: DiseaseValidator = new() {readOnly}

+ «async» GetAll (): Task<IActionResult>  
    {Annotation = HttpGet}  
+ «async» GetById (id : int): Task<IActionResult>  
    {Annotation = HttpGet("{id}")}  
+ «async» Create (diseaseAdd : DiseaseAddDto): Task<IA  
ctionResult>  
    {Annotation = HttpPost}  
+ «async» Update (diseaseUpdate : DiseaseAddDto, id : int):  
Task<IActionResult>  
    {Annotation = HttpPut("{id}")}  
+ «async» Delete (id : int): Task<IActionResult>  
    {Annotation = HttpDelete}

**PatientsController**

«primary» context: HospitalDbContext  
«primary» mapper: IMapper  
- \_validator: PatientValidator = new() {readOnly}

+ «async» GetAll (): Task<IActionResult>  
    {Annotation = HttpGet}  
+ «async» GetById (id : int): Task<IActionResult>  
    {Annotation = HttpGet("{id}")}  
+ «async» Create (patientAdd : PatientAddDto): Task<IA  
ctionResult>  
    {Annotation = HttpPost}  
+ «async» Update (patientUpdate : PatientAddDto, id : int):  
Task<IActionResult>  
    {Annotation = HttpPut}  
+ «async» Delete (id : int): Task<IActionResult>  
    {Annotation = HttpDelete}

**AiController**

«primary» configuration: IConfiguration  
- \_httpClient: HttpClient = new HttpClient  
( ) {readOnly}

+ «async» Predict (image : IFormFile):  
Task<IActionResult>  
    {Annotation = HttpPost("predict")}

**DoctorsController**

«primary» context: HospitalDbContext  
«primary» mapper: IMapper  
- \_validator: DoctorValidator = new() {readOnly}

+ «async» GetAll (): Task<IActionResult>  
    {Annotation = HttpGet}  
+ «async» GetById (id : int): Task<IActionResult>  
    {Annotation = HttpGet("{id}")}  
+ «async» Create (doctorAdd : DoctorAddDto): Task<IAction  
Result>  
    {Annotation = HttpPost}  
+ «async» Update (doctorUpdate : DoctorAddDto, id : int):  
Task<IActionResult>  
    {Annotation = HttpPut}  
+ «async» Delete (id : int): Task<IActionResult>  
    {Annotation = HttpDelete}

**ExaminationsController**

«primary» context: HospitalDbContext  
«primary» mapper: IMapper  
- \_validator: ExaminationValidator = new() {readOnly}

+ «async» GetAll (): Task<IActionResult>  
    {Annotation = HttpGet}  
+ «async» GetById (id : int): Task<IActionResult>  
    {Annotation = HttpGet("{id}")}  
+ «async» Create (examinationAdd : ExaminationAddDto):  
Task<IActionResult>  
    {Annotation = HttpPost}  
+ «async» Update (examinationUpdate : ExaminationAddDto  
, id : int): Task<IActionResult>  
    {Annotation = HttpPut}  
+ «async» Delete (id : int): Task<IActionResult>  
    {Annotation = HttpDelete}

**IdentityController**

«primary» context: HospitalDbContext  
«primary» config: IConfiguration

+ «async» LoginAsync (loginDto : Login  
Dto): Task<IActionResult>  
    {Annotation = HttpPost("login")}  
- GenerateJsonWebToken (doctor : Doctor

					БРУ 10031777.006.005 ПД			
					Диаграмма классов. Контроллеры	Лит.	Масса	Масштаб
Изм	Лист	№ докум.	Подп.	Дата		у		
Разраб.		Пантелеев Г.В.						
Провер.		Выговская Н.В.						
						Лист 1		Листов 2
Реценз.					Дипломный проект	БРУ АСОИ-201		
Н. контр.		Плиско И.Г.						
Утв.		Якимов А.И.						